



Centro Universitário de Brasília – UniCEUB
Faculdade de Tecnologia e Ciências Sociais Aplicadas
Curso de Engenharia da Computação

Rudolf Ullrich Guth

IDENTIFICAÇÃO E PONTO DE PRESENÇA POR ETIQUETA RFID

Brasília
2010

Rudolf Ullrich Guth

IDENTIFICAÇÃO E PONTO DE PRESENÇA POR ETIQUETA RFID

Trabalho de conclusão de curso apresentado à banca examinadora do curso de Engenharia da Computação da FATECS – Faculdade de Tecnologia e Ciências Sociais Aplicadas – Centro Universitário de Brasília – UniCEUB, como requisito para obtenção do título de Engenheiro da Computação.

Orientador: Professor José Julimá Bezerra Júnior

**Brasília
2010**

Este Trabalho foi julgado adequado para a obtenção do Título de Engenheiro de Computação, e aprovado em sua forma final pela Faculdade de Tecnologia e Ciências Sociais Aplicadas – FATECS.

Prof. Abiezer Amarilia Fernandez
Coordenador do Curso

Banca Examinadora:

Prof. José Julimá Bezerra – Mestre em Engenharia Elétrica.
Instituto Militar de Engenharia - RJ
Orientador

Prof. Luís Cláudio Lopes de Araújo – Mestre em Matemática Pura
Universidade de Brasília - DF

Prof. Vera Farini – Mestre em Matemática.
Universidade de Brasília - DF

Prof. Francisco Javier de Obaldia Diaz – Mestre em Engenharia Elétrica.
Universidade de Brasília - DF

DEDICATÓRIA

Dedico este trabalho a todas as pessoas que sempre estiveram ao meu lado, me apoiando nesta jornada para a minha graduação como Engenheiro de Computação.

AGRADECIMENTO

Agradeço a toda minha família pelo apoio, compreensão e ajuda. Ao meu pai Francisco, por ser o exemplo de homem que sempre seguirei em minha vida e à minha mãe Cândida por educar-me e cuidar de mim sempre que precisei e, principalmente, agradeço a eles por me darem uma vida maravilhosa, cheia de alegrias e de oportunidades. À minha irmã Flávia por sempre dar-me carinho e conselhos. À Paula pelo companheirismo, apoio incondicional, carinho e conselhos nas horas difíceis. Aos meus sobrinhos por me proporcionarem alegrias das mais variadas. Aos meus irmãos de sangue, Thomé e Thiago, pela amizade e respeito. Aos irmãos da vida, amigos do G10, que sempre estiveram ao meu lado apoiando-me e alegrando-me. Ao José Carlos, monitor de circuitos, que muito me auxiliou no desenvolvimento do meu circuito.

RESUMO

Busca-se, com este projeto, mostrar a importância e a necessidade das empresas se sentirem seguras em relação ao acesso de pessoas em suas dependências. Para tanto, faz-se necessário um mecanismo de controle deste acesso, mas que se mostre prático e que a relação custo benefício seja o indicador que servirá de elo entre os benefícios e as necessidades da implantação do projeto. Desenvolveu-se, sob esta ótica, o presente trabalho monográfico, cujo objetivo é mostrar como e porque agilizar e controlar o acesso de pessoas às dependências de uma empresa, garantindo mais segurança à empresa. O aspecto mais relevante do projeto consiste na ideia do uso de um cartão RFID que contenha as informações pessoais e profissionais dos funcionários. Com estas informações, quando o funcionário aproximar o cartão de uma leitora ele estará se identificando para o banco de dados da empresa que, ao reconhecê-lo, efetuará a abertura de uma fechadura eletrônica e o acendimento de um LED de cor verde, o que fará com que o ponto de entrada ou de saída da pessoa seja marcado. Caso seja recusado o cartão um LED de cor vermelha acenderá, indicando a recusa. Para esta solução, um banco de dados MySQL é utilizado como armazenador da identificação do cartão e dos horários e data de entrada e de saída do usuário. Foi desenvolvido, para o correto funcionamento, um algoritmo que interpreta a informação dada pelo computador ao circuito eletrônico e efetua a abertura da fechadura e o acendimento dos LEDs. Juntando todas as partes pode-se disponibilizar a agilidade e a segurança propostas inicialmente pelo projeto.

Palavras-chave: RFID, Controle de acesso, Circuito Eletrônico, LED, Banco de dados.

ABSTRACT

With this project, the importance and need for companies to feel secure about the access of people in their dependencies is shown. Therefore, it is necessary a control mechanism for this access, which can show that it can be practical and cost effective, which it is the indicator that will serve as a link between benefits and needs of implementation of the project. Developed, from this perspective, this monograph, pretends to show how and why accelerate and control the access of people to the premises of a company, can ensure more security to the company. The most relevant aspect of the project consists of the idea of using an RFID tag that contains personal and professional information. With this information, when an employee brings a card to a reader, to identified himself with the company's database, and if recognized, will open an electronic lock and a green LED lights up, which will cause the entry point or leaving of the person to be marked. If the card is refused a red LED will light, indicating the refusal. For this solution, a MySQL database is used as storage of the identification card and timetables and entry and exit of the user. Was developed for the correct operation, an algorithm that interprets the information given by the computer to the electronic circuit to open the lock and firing of the LEDs. Bringing together all the parties can provide the speed and security initially proposed by the project.

Key-words: RFID, Access Control, Electronic Circuit, LED, Database.

SUMÁRIO

LISTA DE FIGURAS.....	ix
LISTA DE QUADROS	x
LISTA DE SIGLAS E ABREVIACÕES	xi
INTRODUÇÃO	1
MOTIVAÇÃO	1
Objetivos.....	2
Estrutura do Projeto Monográfico	3
1 IDENTIFICAÇÃO POR RÁDIO FREQUÊNCIA – RFID.....	4
1.1 Origem	4
1.2 Conceitos Básicos sobre RFID.....	5
1.2.1 Frequência de transmissão e de recepção	6
1.2.2 Componentes do RFID	8
1.2.2.1 Etiqueta, tag ou transponder	8
1.2.3 Técnicas de comunicação	9
1.2.4 Leitoras de etiquetas	10
1.2.5 Distância da leitura	11
1.2.6 Antena RFID	12
1.2.7 Problemas com Interferências.....	12
1.2.8 RFID e o Código de Barras.....	12
1.2.9 Vantagens, Benefícios e Desvantagens do RFID	13
1.2.10 Casos de Uso do RFID	14
2 REFERENCIAL TECNOLÓGICO	16
2.1 Linguagens de Programação	16
2.1.1 Linguagem PHP.....	16
2.1.2 Linguagem C	17
2.2 Banco de Dados MySQL	18
2.3 Porta Paralela	18
2.3.1 Endereçamento de Porta Paralela	19
2.3.2 Conector DB25.....	20
2.3.2.1 Pinagem.....	21
2.4 Porta Serial.....	23
2.5 Fechadura Eletromagnética	24
3 IDENTIFICAÇÃO E PONTO DE PRESENÇA POR ETIQUETA RFID	25
3.1 Ordem de Criação e Planejamento do Projeto	25
3.1.1 Estrutura Geral do Projeto.....	26
3.1.2 Funcionamento Básico do Projeto.....	26
3.2 Dispositivos Eletrônicos do Projeto	28
3.2.1 Especificações dos dispositivos utilizados no Projeto	30
3.2.1.1 Leitora RFID AP-09 Acura Technologies.....	30
3.2.1.2 Circuito Eletrônico	31
3.2.1.3 Fechadura Eletromagnética	32
3.3 Software.....	33
3.3.1 Principais Funções Utilizadas na Programação do Projeto	34
3.3.1.1 Porta Paralela DB25.....	34

3.3.1.2	RFID.....	34
3.3.1.3	Banco de Dados.....	34
3.4	Testes e Resultados	35
3.5	Simulação.....	40
4	CONCLUSÃO.....	43
4.1	Problemas Encontrados	44
4.2	Sugestões de Trabalhos Futuros.....	45
	REFERÊNCIAS	46
	ANEXO A: CÓDIGO FONTE COMUNICAÇÃO COM A PORTA PARALELA.	48
	ANEXO B: PROCESSA PONTO	52
	ANEXO C: CADASTRA, ALTERA E EXCLUI DADOS DE FUNCIONÁRIOS.....	56
	ANEXO D: RELATÓRIO DE PONTOS.....	61
	ANEXO E: RELATÓRIO POR FUNCIONÁRIO	66
	ANEXO F: RELATÓRIO POR DATA	69
	ANEXO G: MENU.....	72
	ANEXO H: MARCAÇÃO DO PONTO DO FUNCIONÁRIO.....	73
	ANEXO I: CONEXÃO E DESCONEXÃO COM O BANCO DE DADOS.....	77

LISTA DE FIGURAS

FIGURA 1.1 - TOPOLOGIA DE LEITURA.	5
FIGURA 1.2 – TAG, ETIQUETA.	6
FIGURA 1.3 – TABELA DE FREQUÊNCIAS	7
FIGURA 1.4 – COMPONENTES DE UMA ETIQUETA.	8
FIGURA 1.5 – SEQUÊNCIA DE LEITURA	10
FIGURA 2.1 – PORTA PARALELA DO COMPUTADOR	21
FIGURA 2.2 – CONECTOR MACHO DO CABO PARALELO	21
FIGURA 2.3 – PINAGEM DA PORTA PARALELA	22
FIGURA 2.4 – DESCRIÇÃO DA PINAGEM DA PORTA PARALELA	23
FIGURA 2.5 – FECHADURA ELETROMAGNÉTICA	24
FIGURA 3.1 – ORDEM DE PLANEJAMENTO DO PROJETO	25
FIGURA 3.2 – FLUXOGRAMA DO FUNCIONAMENTO DO PROJETO	27
FIGURA 3.3 – CIRCUITO ELETRÔNICO	28
FIGURA 3.4 – MAPA DO CIRCUITO E COMPONENTES	29
FIGURA 3.5 – MAPA DE TRILHAS DO CIRCUITO	29
FIGURA 3.6 – LEITORA RFID AP-09.....	31
FIGURA 3.7 – FECHADURA ABERTA	33
FIGURA 3.8 – FECHADURA FECHADA	33
FIGURA 3.9 – TESTE CIRCUITO ACESSO PERMITIDO	35
FIGURA 3.10 – TESTE CIRCUITO ACESSO NEGADO.....	36
FIGURA 3.11 – TESTE ABERTURA PORTA COM CIRCUITO	36
FIGURA 3.12 – TELA MENU	37
FIGURA 3.13 – TELA INCLUSÃO DE FUNCIONÁRIOS.....	37
FIGURA 3.14 – TELA ALTERAÇÃO DE CADASTRO	38
FIGURA 3.15 – TELA MENU DE RELATÓRIO.....	38
FIGURA 3.16 – FOLHA DE PONTO POR NOME	39
FIGURA 3.17 – FOLHA DE PONTO POR DATA	39
FIGURA 3.18 – CARTÃO IDENTIFICADO E PONTO MARCADO	40
FIGURA 3.19 – PORTA E FECHADURA	41
FIGURA 3.20 – CIRCUITO DO PROJETO.....	41
FIGURA 3.21 – LEITORA RFID <i>ACURA TECNOLOGIES</i>	41
FIGURA 3.22 – ADAPTADOR TTL RS 232	42

LISTA DE QUADROS

QUADRO 1.1 – FREQUÊNCIAS E DISTÂNCIAS.....	11
QUADRO 1.2 – COMPARAÇÃO DE CÓDIGO DE BARRAS E <i>TAGS</i> RFID	13
QUADRO 2.1 – ENDEREÇOS DE PORTA PARALELA.....	20
QUADRO 3.1 – DESCRIÇÃO DA UTILIZAÇÃO DOS PINOS DA PORTA PARALELA.....	30

LISTA DE SIGLAS E ABREVIACÕES

RFID - *Radio Frequency Identification* – Identificação por Rádio Frequência

CPD – Centro de Processamento de Dados

LED – *Light Emitting Diode* – Diodo Emissor de Luz

IFF – *Identify Friendly or Foe* – Identificar Aliado ou Inimigo

IBM – *International Business Machines*

UHF – *Ultra High Frequency* – Frequência Ultra Alta

ID – Identificador

ISM – *Industrial Scientific Medical* – Industrial Científica Médica

VLF – *Very Low Frequency* - Frequência Muito Baixa

LF – *Low Frequency* - Frequência Baixa

MF - *Medium Frequency* - Frequência Média

HF - *High Frequency* – Frequência Alta

VHF – *Very High Frequency* - Frequência Muito Alta

SHF – *Super High Frequency* – Frequência Super Alta

EHF – *Extremely High Frequency* – Frequência Extremamente Alta

GPS – *Global Positioning System* – Sistema de Posicionamento Global

RLTS - *Radio Linked Telemetry System* – Sistema de Telemetria por Conexão de Rádio

PHP – *Personal Home Page* – Pagina Inicial Pessoal

GPL – *General Public License* – Licença Pública Geral

SGBD – Sistema de Gerenciamento de Banco de Dados

SQL – *Structured Query Language* – Linguagem Query Estruturada

SPP – *Standard Parallel Port* – Porta Paralela Padrão

CPU – *Central Processing Unit* – Unidade de Processamento Central

EPP – *Enhanced Parallel Port* – Porta Paralela Reforçada

ECP – *Enhanced Capabilities Port* – Porta de Capacidades Reforçada

DMA – *Direct Memory Access* – Acesso de Memória Direto

FIFO – *First In First Out* – Primeira Entrada Primeira Saída

LPT – *Line Print Terminal* – Linha de Terminal de Impressão

GND – *Ground* – Terra

USB – *Universal Serial Bus* – Porta Serial Universal

PDA – *Personal Digital Assistant* – Assistente Pessoal Digital

TTL – *Transistor Transistor Logic* – Lógica Transistor Transistor

ASCII – *American Standard Code II* – Código Americano Padrão II

RS 232 – *Recommended Standard 232* – Padrão Recomendado 232

NPN – Negativo Positivo Negativo

PNP – Positivo Negativo Positivo

PN – Positivo Negativo

DLL – *Dynamic Link Library* – Biblioteca de Conexão Dinâmica

INTRODUÇÃO

Este trabalho de conclusão do curso de Engenharia de Computação propõe um identificador e ponto de presença por RFID (*Radio Frequency Identification* – Identificação por Rádio Frequência). A motivação e os objetivos deste projeto são apresentados neste capítulo, junto à estrutura da monografia.

MOTIVAÇÃO

O avanço da tecnologia cria ferramentas e métodos que podem ser utilizados para facilitar a rotina das pessoas. E é em razão disso que a popularização da automatização de residências e de empresas leva à reflexão e à reavaliação desta rotina. Atualmente, tarefas como utilizar a chave em fechadura podem não ser mais necessárias em diversos lugares, os quais, às vezes, não necessitam que nada seja inserido nas fechaduras. Utilizando a tecnologia disponível, pode-se automatizar muito mais que apenas fechaduras de portas, como, por exemplo, cortinas, o acendimento de lâmpadas, entre outras atividades. Estes avanços na tecnologia permitem, hoje, que o controle remoto seja muito utilizado, e que tarefas administrativas de empresas possam ser realizadas com o auxílio de um computador. Em vista disto, propõe-se, com este projeto monográfico, maior agilidade na identificação e no controle de entrada e de permanência de pessoas nas dependências de uma empresa.

A chave ainda é a ferramenta mais comum para que funcionários entrem nas dependências das empresas, enquanto cadernos ou folhas de papel são usados na marcação do ponto de trabalho de cada um deles. Esta forma de registro do ponto de presença torna-se bastante sujeita a erros humanos, podendo incorrer em marcação indevida de horários de chegada e de saída dos funcionários, tornando-se, assim, um controle de horas bastante confuso e impreciso. Há, ainda, um problema mais sério, que é o baixo controle da entrada de pessoas em determinados lugares, cuja maior necessidade é exatamente controlar este fluxo. São, por exemplo, áreas financeiras ou de acesso a documentos confidenciais de empresas e, também, em Centros de Processamento de Dados – CPDs.

Na tentativa de eliminar os riscos relativos à segurança de partes importantes ou até mesmo da completa segurança de todas as dependências da empresa é que o sistema de identificação e de ponto de presença por RFID sugere a união entre a marcação dos pontos e o controle do acesso de pessoas. Com um sistema conjunto é possível obter mais segurança

para informações críticas que podem ser encontradas em setores que necessitam de mais atenção quanto a evitar vazamento indevido destas informações, sobretudo as confidenciais, para pessoas externas à empresa.

Há ainda mais vantagens ao se utilizar um sistema automatizado, quando, por exemplo, a ausência da interferência humana no processo, diminui o risco de distorção em relação às horas de trabalho exercidas pelos funcionários. É fato, no entanto, que podem ainda existir vulnerabilidades, mas este risco é reduzido quando o número de pessoas envolvidas no processo é bem menor.

Assim sendo, o sistema de identificação e de ponto de presença por RFID pode ser visto como nova forma de administrar o banco de horas dos funcionários e como eficiente controle das partes internas de uma companhia, utilizando-se recursos tecnológicos que mostrarão um método diferente destas funcionalidades administrativas.

Objetivos

Objetivo Geral

Criação de solução visando a administração mais eficiente no que se refere ao controle de entrada e de deslocamento de pessoas dentro de uma empresa e também a um sistema que efetue maior controle do banco de horas, auxiliando, assim, o Setor de Recursos Humanos, na administração das horas trabalhadas pelos funcionários.

Objetivos Específicos

- Criar um banco de dados que registre corretamente o horário de entrada e de saída dos funcionários.
- Organizar um banco de dados com informações relativas aos funcionários e aos setores onde trabalham.
- Demarcar o horário de entrada e de saída dos funcionários.
- Efetuar a abertura da fechadura da porta, por meio eletrônico, e o acendimento dos LEDs (Light Emitting Diode – Diodo Emissor de Luz) que acusam se a identificação foi aceita ou recusada.
- Receber o sinal de um sensor de abertura de porta para que o ponto seja realmente marcado à entrada do funcionário.

Estrutura do Projeto Monográfico

Esta monografia está dividida em quatro capítulos, sendo que a introdução constitui o primeiro e os subsequentes estão divididos da seguinte forma:

Capítulo 1 – Identificação por Rádio Frequência - RFID

Nesse capítulo é apresentada toda a consulta feita em relação ao histórico do RFID, seus componentes e forma de funcionamento, tal como interferências e comparações com código de barras.

Capítulo 2 – Referencial Tecnológico

Nesse capítulo é apresentado o funcionamento do banco de dados MySQL, introdução aos conceitos das linguagens de programação. Também é apresentado aos conceitos da porta paralela e serial e como é o mecanismo da fechadura eletromagnética.

Capítulo 3 – Identificação e ponto de presença por etiqueta RFID

Esse capítulo descreve a solução proposta neste projeto como a arquitetura do sistema, seu desenvolvimento e as soluções sugeridas. Resultados e testes também são apresentados neste capítulo.

Capítulo 4 – Conclusão

A conclusão apresenta as principais constatações feitas durante o desenvolvimento do sistema de identificação e o ponto de presença por RFID, pré-requisitos para o correto funcionamento. Sugestões de novos projetos são mostradas nesse capítulo, além de apresentados os problemas que foram encontrados durante o desenvolvimento do projeto.

1 IDENTIFICAÇÃO POR RÁDIO FREQUÊNCIA – RFID

Neste capítulo é abordada a tecnologia RFID utilizada no projeto para a leitura dos cartões, através da leitora, responsáveis pela identificação das pessoas que foram cadastradas previamente no banco de dados, para que a entrada delas em determinadas dependências seja permitida ou negada.

1.1 Origem

O primeiro modo de RFID – *Radio Frequency Identification*, foi utilizado na Segunda Guerra Mundial, tendo em vista que tanto os alemães quanto os japoneses e os americanos precisavam identificar, com antecedência, a aproximação de aviões amigos ou inimigos. No entanto, havia um grande problema, pois ainda não era possível identificar, por meio destes radares, quais aviões estavam se aproximando. Em vista disso, os alemães desenvolveram uma técnica que, no caso de seus aviões girarem, o sinal recebido pelo radar, nas bases, seria diferente e, assim, os responsáveis poderiam identificá-los como aviões alemães. Dessa forma, considerou-se este como o primeiro tipo e também como a primeira utilização do sistema passivo de RFID. Após este tipo de utilização, o escocês Sir Robert Alexander Watson-Watt, considerado inventor do radar, dele tomou conhecimento e desenvolveu para os britânicos um novo tipo de radar facilitando a identificação de aviões. Sir Robert desenvolveu um transmissor para que quando recebessem sinais dos radares, os transmissores dos aviões enviassem uma resposta de *amigo* para ser retransmitida para a base e, assim, os radares serem identificados pelos operadores. Assim, o projeto IFF – *Identify Friendly or Foe* foi criado pelos britânicos para que identificassem seus aliados com mais clareza e velocidade. É interessante lembrar que, ainda hoje, o controle aéreo dos países é pautado nesse princípio que Watson-Watt utilizou na Segunda Guerra Mundial (RFID Journal - RFID, 2010).

Após a guerra, cientistas começaram a desenvolver pesquisas acerca do RFID, visando utilização em meios civis, mas até o final dos anos 60 eram usados na identificação de áreas nucleares, pelos militares. Posteriormente, no entanto, companhias aprimoraram sistemas para a prevenção de furtos em lojas, fazendo uso das *tags* (etiquetas), as quais ainda hoje são utilizadas (RFID Journal - RFID, 2010).

Em 1973, Mario W. Cardullo requereu a primeira patente do sistema de RFID para etiquetas ativas, enquanto Charles Walton requeria a patente para uma etiqueta passiva.

Charles Walton valia-se da etiqueta para abrir as portas sem o uso de chaves (RFID Journal - RFID, 2010).

Nos anos 90, a empresa IBM começou seu desenvolvimento com a tecnologia RFID. Com isso, aprimoraram e patentearam um sistema utilizando tecnologia UHF – *Ultra High Frequency*, tornando a distância para a leitura e transferência de dados maior e mais eficaz e, assim, aumentando a velocidade de identificação de produtos, como por exemplo, em empresas de estoque. Após o fracasso da negociação do sistema com o Walmart, a IBM vendeu sua patente à Intermec, que incorporou-a ao seu sistema de leitura de código de barras (RFID Journal, 2010).

1.2 Conceitos Básicos sobre RFID

O RFID é a tecnologia que pode ser aplicada de diversas formas e que leva esse nome por ser um sistema de identificação por Rádio Frequência e a sigla é a abreviação de *Radio Frequency Identification*, em inglês. O funcionamento básico desse sistema é o armazenamento de informações transmitidas por ondas de rádio para a comunicação entre os dispositivos, sendo que essas informações ficam nas *tags* RFID. Cada dispositivo contém informações e, ao se utilizar suas antenas o sinal é transmitido e o sistema de leitura é acessado (Figura 1.1), fazendo com que as informações sejam colhidas pela leitora (Congresso RFID, 2010).



Figura 1.1 - Topologia de leitura.
Fonte: www.softwareadvice.com

Esses dispositivos são conhecidos como *TAG* RFID, *CARTÃO* RFID ou *ETIQUETA* RFID (Figura 1.2), os quais são um *transponder* – sigla de *Transmitter-responder* e contêm antenas e *chips* de silício que permitem responder aos sinais de rádio enviados de uma base transmissora. Maior abrangência quanto às *tags* é feita em detalhes no item 1.2.2.1.

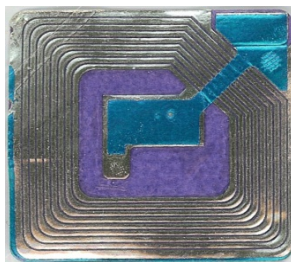


Figura 1.2 – TAG, Etiqueta.

Fonte: www.softwareadvice.com

Em razão disso, quando o sinal é emitido e captado pela antena o sistema de leitura é acessado, mostrando toda a informação contida no dispositivo. Este dispositivo pode ser um identificador (ID), ou um conjunto de informações relativas à *tag* RFID. Em sistemas robustos é capturado o ID da *tag*, o qual é utilizado para acessar o conjunto de informações por meio de um banco de dados remoto.

1.2.1 Frequência de transmissão e de recepção

Como o próprio nome diz, o RFID funciona com trocas de informações através de frequências de rádio, portanto, faz-se necessário utilizar-se de uma faixa de frequência previamente determinada para que não haja nenhuma forma de interferência quando a troca de dados ocorrer e, em razão disso, dificultar a leitura das *tags*; e também para que o sistema não interfira no funcionamento de outros equipamentos que estejam por perto.

Ao se desenvolver um sistema que utiliza a tecnologia de RFID deve-se levar em conta a frequência utilizada por outros sistemas, e é por isso que existem frequências estabelecidas para diversas aplicações, tais como industriais, científicas e médicas, conhecidas como faixa de frequência ISM – *Industrial Scientific Medical* (Congresso RFID, 2010).

Na figura 1.3 retirada do *site* Congresso RFID, são mostradas faixas de frequências que devem estar contidas no desenvolvimento de sistemas RFID.

	VLF	LF	MF	HF	VHF	UHF	SHF	EHF	
f	0,01	0,1	1	10	100	1000	10000	100000	Mhz
λ	30000	3000	300	30	3	0,3	0,03	0,003	m

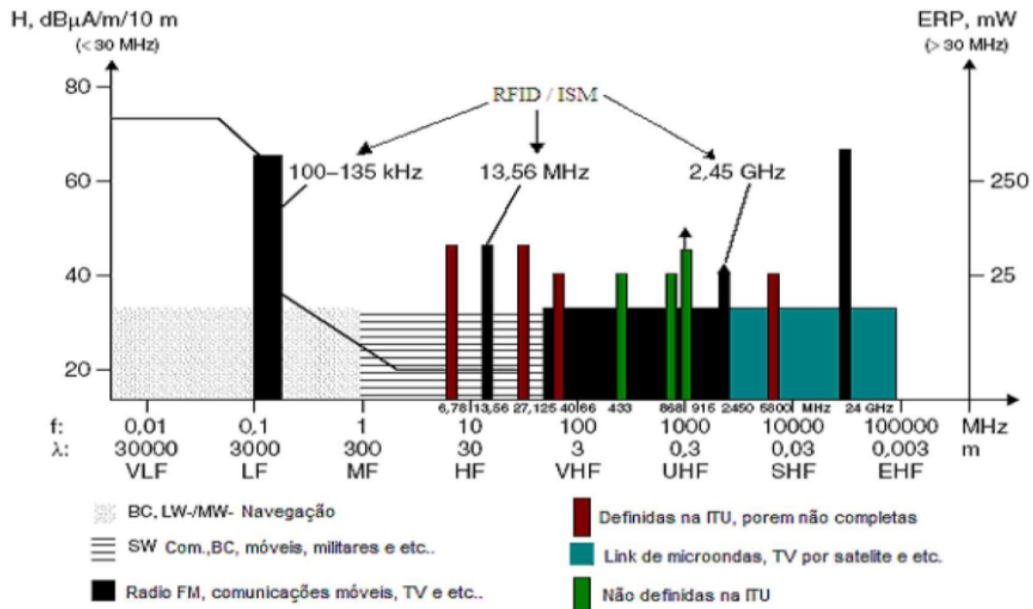


Figura 1.3 – Tabela de frequências

Fonte: www.congressorfid.com.br

Copiado do *site* Congresso de RFID, o gráfico constante da Figura 1.3 mostra a divisão entre os tipos de frequência, indicando seu valor de f (frequência), λ (Lambda) que é o comprimento da onda, e sua classificação. VLF é *Very Low Frequency* (Frequência Muito Baixa), LF é *Low Frequency* (Frequência Baixa), MF é *Medium Frequency* (Frequência Média), HF é *High Frequency* (Frequência Alta), VHF é *Very High Frequency* (Frequência Muito Alta), UHF é *Ultra High Frequency* (Frequência Ultra Alta), SHF é *Super High Frequency* (Frequência Super Alta), EHF é *Extremely High Frequency* (Frequência Extremamente Alta).

As principais faixas de frequências são:

– *Low Frequency* (Baixa Frequência – LF) – Faixa de operação de 125 kHz até 134 kHz. São ondas de longo alcance podendo chegar a 1000 km e de baixo custo de operação. Nesta faixa o principal uso é para equipamentos militares e marítimos (Congresso RFID).

– *High Frequency* (Alta Frequência – HF) – faixa de operação de 13,56 MHz para equipamentos RFID. Nesta faixa de frequência podem ser realizadas conexões transcontinentais, mas ela pode ser também utilizada por sistemas de controle remoto e sistemas de RFID indutivo (Congresso RFID, 2010).

– *Ultra High Frequency* (Ultra Alta Frequência – UHF) – faixa de operação de 860 MHz até 960 MHz. Frequências que podem ser utilizadas por SRDs (*Short Range Devices*),

tais como equipamentos de curto alcance e portões automáticos. Vale ressaltar que fora da Europa esta faixa pode ser utilizada para equipamentos RFID (Congresso RFID – Frequências, 2010).

1.2.2 Componentes do RFID

Um sistema genérico RFID é composto por etiquetas inteligentes (também conhecidas como *tag* RFID ou *transponders*), por um ou mais leitores (também conhecidos como interrogadores ou *transceptors*), por uma antena ou bobina e por um sistema computacional que utiliza os dados captados pelos leitores RFID (THORNTON, 2006).

1.2.2.1 Etiqueta, *tag* ou *transponder*

Uma etiqueta nada mais é que um *microchip* que possui uma antena para que a comunicação entre ela e a leitora seja feita, tanto recebendo sinal quanto enviando sinal. O sinal emitido pela *tag*, basicamente, possui sua identidade, muitas vezes chamada apenas de ID, sendo que algumas contêm informações adicionais como identificação de algum funcionário ou cliente. As etiquetas podem ter diversos formatos, desde um simples adesivo a um cartão plástico como um crachá. Mas todos funcionam utilizando o mesmo princípio. A Figura 1.4 ilustra uma etiqueta e seus componentes.

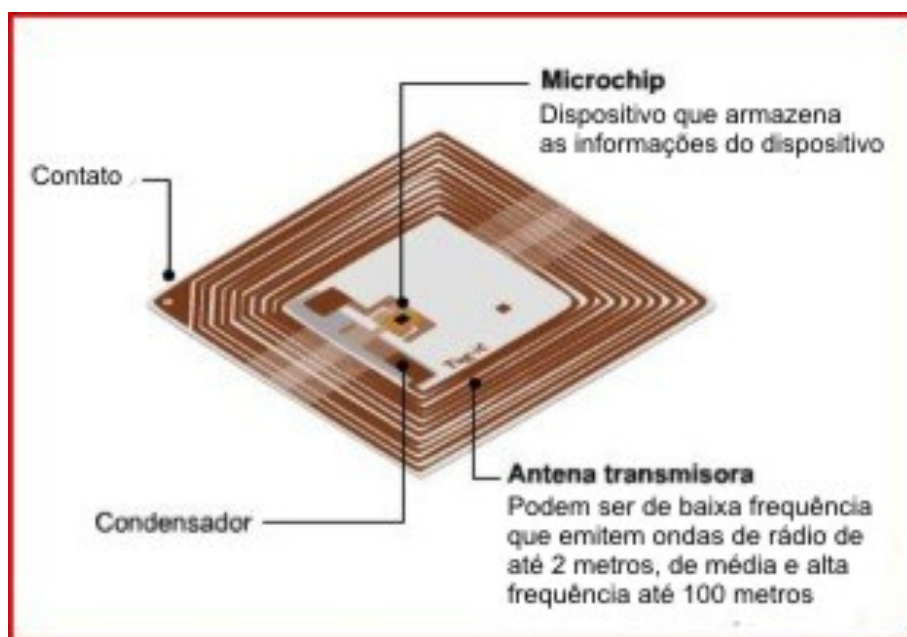


Figura 1.4 – Componentes de uma Etiqueta.

Fonte: <http://leonardo4c.files.wordpress.com/2009/04/rfid.jpg>

A frequência de operação e o tipo de etiquetas utilizadas fazem bastante diferença em relação ao desempenho de todo o sistema. O tipo de etiqueta é determinado por dois fatores: se a etiqueta é capaz de iniciar a comunicação e se a etiqueta tem fonte de energia própria. Pautadas nesses dois fatores existem três tipos de etiquetas: passivas, semipassivas e ativas (SANGHERA, 2007).

Etiquetas passivas são aquelas que não possuem nem emissão nem fonte de energia próprias, ou seja, necessitam das ondas de rádio enviadas pela leitora para transformá-las em energia e, assim, enviar as informações gravadas no *microchip*. Elas refletem o sinal enviado pelo leitor. É interessante ressaltar que o preço desse tipo de etiqueta pode variar de acordo com a capacidade que ela tiver de armazenamento e de fabricação. Estas *tags* são lidas a uma distância de 10 mm, podendo chegar a 6 metros e têm vida útil longa por não precisarem de bateria.

Etiquetas semipassivas utilizam baterias para que o *microchip* instalado possa funcionar, mas usam a energia das ondas de rádio enviadas pelo leitor para também poder enviar as informações nelas contidas. Algumas dessas etiquetas podem ficar em modo *stand by* para economizar bateria. O alcance para este tipo de etiqueta pode chegar a 30 metros, mas sua vida útil é menor se comparada à etiqueta passiva.

Etiquetas ativas precisam de baterias e podem até utilizar a luz solar como fonte de energia. Este tipo de *tag* é capaz de enviar os dados de forma própria, pois contém nela um transmissor e, assim, não necessita esperar por um pedido do leitor, ao contrário da etiqueta passiva. É tipo de etiqueta mais aproveitado para rastrear objetos em longos prazos ou de grande importância como, por exemplo, no exército norte-americano que rastreia *containers* com equipamentos e armas. Por serem capazes de enviar de forma própria suas informações, as etiquetas ativas transmitem a uma distância que pode chegar a centenas de metros e, além disso, têm maior capacidade de armazenamento de informações. Etiquetas ativas podem ser utilizadas em equipamentos que possuem GPS – *Global Positioning System* para identificar, por exemplo, um caminhão de determinada empresa em uma estrada, mas seu tempo de vida útil é o mais baixo entre todas, em função de maior utilização das baterias.

1.2.3 Técnicas de comunicação

Para se determinar a faixa de leitura de um sistema RFID, precisa-se primeiro saber qual a técnica de comunicação e a potência da antena utilizadas e, por fim, qual o tipo de etiqueta que está sendo usada. De acordo com o livro *RFID+ Study Guide and Practice*

Exam de Paul Sanghera, existem apenas duas técnicas de comunicação que são o acoplamento indutivo e acoplamento difuso de retorno (SANGHERA, 2007).

No acoplamento indutivo tanto o leitor quanto a etiqueta utilizam bobinas como antenas. Essas bobinas criam campos magnéticos, sendo que a variação, no campo magnético, serve para transferir potência (e dados) entre o leitor e as etiquetas. É uma técnica que limita a área de leitura porque ela só funciona no campo de atuação das bobinas. Dessa forma, o acoplamento indutivo requer que o leitor esteja próximo da etiqueta. Com isso, a distância de leitura é por volta de 30 cm para LF (Frequências Baixa) e de 1m para HF (Frequências Altas) (SANGHERA, 2007).

O Acoplamento Difuso de Retorno geralmente é utilizado por etiquetas passivas operando em UHF – Frequências Ultra Altas ou microondas. Como este acoplamento trabalha além do campo de atuação das bobinas, ele permite maiores áreas de leitura. A maior área de atuação conseguida com esse tipo de acoplamento é devido ao uso de ondas eletromagnéticas para estabelecer a comunicação, em vez do campo magnético utilizado pelo acoplamento indutivo (SANGHERA, 2007).

1.2.4 Leitoras de etiquetas

Muitas vezes chamada de interrogador, por interrogar as etiquetas, a leitora pode ter uma ou mais antenas para a emissão de ondas de rádio e para receber o sinal enviado pelas *tags*. Pode conseguir “ler” estas etiquetas através de diferentes tipos de matérias, tais como madeira, cimento, vidro e plástico. Após a decodificação do sinal recebido através do *transceiver*, o leitor envia as informações recebidas das etiquetas para um computador ou servidor para que ele possa processá-las e utilizá-las para diferentes aplicações, como ilustrado na Figura 1.5 (Saber Eletrônica, 2010).

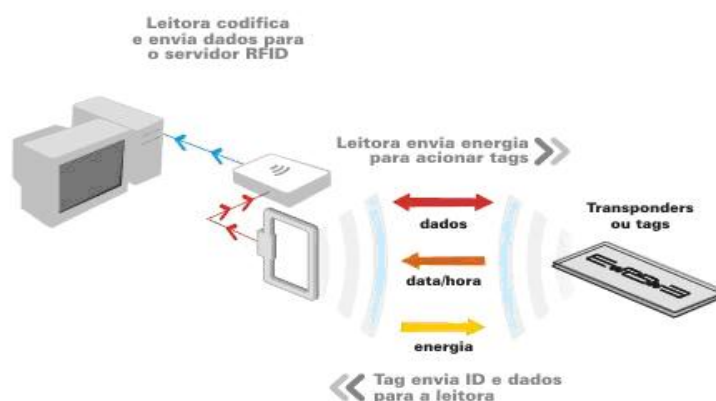


Figura 1.5 – Sequência de leitura

Fonte: Sinal Amplificado – Telecomunicações, Redes, Móviles, TI.

Basicamente, as leitoras são reunidas em apenas dois grupos: somente Leitura (*Read-only*) e Leitura e Escrita (*Read and write*).

Somente Leitura: apenas “lê” a informação contida nas etiquetas, não podendo alterar, nela, nenhum dado.

Leitura e Escrita: são leitoras que podem tanto ler as informações quanto alterar as informações contidas nas etiquetas, mas somente podem escrever em etiquetas com a opção de escrita.

1.2.5 Distância da leitura

A distância da leitura pode ser influenciada por diversos fatores tais como o tipo da etiqueta, a potência da leitora, o tamanho da antena tanto da *tag* quanto da leitora e a direção para a qual estão sendo apontadas. Para que seja possível determinar a distância que a aplicação está usando é necessário, primeiramente, saber a frequência em que as etiquetas estão operando. Sendo assim, a distância pode variar de sistema para sistema, como é informado no Quadro 1.1.

Quadro 1.1 – Frequências e distâncias.

FREQUÊNCIA UTILIZADA	DISTÂNCIA	OBSERVAÇÕES
125 kHz a 134.3 kHz (LW)	Até 30 cm	Geralmente é utilizada em cerca de 10 cm.
13.56 MHz (HF)	Até 1,5 m	Geralmente é utilizada em cerca de 1 m.
860 MHz a 960 MHz (UHF)	Mínimo de 1 m e máximo de 12 m	Podendo chegar a 15 m dependendo do <i>microchip</i> .
RLTS	23500 metros quadrados	Sistema de Localização em Tempo Real
2.45 GHz (SHF)	Mais que 100 metros	Pode-se ter informações precisas de localização da <i>tag</i>

1.2.6 Antena RFID

Antena RFID é a parte do sistema que é responsável por emitir e receber os sinais das etiquetas. Etiquetas passivas geralmente têm uma antena enrolada em sua extensão, a qual combina com a antena da leitora e assim formam um campo magnético. Tem-se diversos tipos e tamanhos tanto para as leitoras como para as etiquetas. Quando as etiquetas possuem antenas acopladas ao seu corpo, estas antenas estão diretamente conectadas ao *microchip*. As antenas das leitoras emitem ondas de rádio. A energia enviada por rádio frequência da leitora é de certa forma colhida pela antena da *tag* para que os dados sejam enviados de volta à leitora.

1.2.7 Problemas com Interferências

Existem dificuldades de aplicação de sistemas que utilizam RFID devido às superfícies nas quais as *tags* são instaladas. Superfícies como metal, água, sucos e gelo, podem atrapalhar na hora da troca de informações entre a leitora e a etiqueta, o que acontece principalmente quando a taxa de transferência está alta, ou seja, quanto mais alta a frequência utilizada nas *tags* para este tipo de material, maior é a interferência causada. Há também interferências com redes sem fio, o que ocorre apenas em algumas frequências, tendo em vista que as frequências utilizadas por redes sem fio nem sempre são as mesmas do RFID e, com isso, apenas determinadas redes podem sofrer influência na hora da transmissão de dados. Mas a recepção de sinal pelas leitoras ou antenas depende não somente do material utilizado, mas também pelo ângulo no qual as antenas estão sendo usadas.

1.2.8 RFID e o Código de Barras

Muitas pessoas fazem a comparação entre o RFID e o Código de Barras, principalmente por entenderem que o RFID veio para atualizar e substituir.

O Código de Barras tem algumas desvantagens quanto a sua leitura o que, no caso do RFID, o processo é mais rápido, pois não precisa estar, de certa forma, alinhado à leitora. Sem contar que as etiquetas podem ter mais dados armazenados em sua memória, além de não ser necessário que algum operador aponte algo para as *tags*, pois estando dentro do raio de leitura, ela pode ser feita de forma automática. A comparação também pode ser feita como no Quadro 1.2.

Quadro 1.2 – Comparação de Código de Barras e *TAGS* RFID

CÓDIGO DE BARRAS	TAGS RFID
A leitura requer contato	Leitura sem linha de vista
Só contém um código	Identifica cada produto individualmente
Leituras sequenciais e com intervenção humana	Leituras múltiplas, simultâneas e de forma automática
Mesmo código para produtos do mesmo tipo	Código único, próprio de fábrica ou escrito a distância
Degradam facilmente	Resistentes a longo prazo

1.2.9 Vantagens, Benefícios e Desvantagens do RFID

Como toda tecnologia emergente e em desenvolvimento, o RFID tem vantagens, benefícios e desvantagens quanto a sua utilização nas mais variadas formas de aplicação.

A rápida leitura e a identificação dos cartões são grandes vantagens que o RFID apresenta, tal como a minimização de erros quanto à identificação de produtos, de pessoas. Pode-se também adicionar a esta agilidade do controle o rápido armazenamento de dados obtidos pelas *tags*. Outra grande vantagem é a independência do ambiente utilizado, ou seja, podem ser ambientes empoeirados, chuvosos e até meio corrosivos, uma vez que não precisa de contato direto, não há necessidade de interferência humana para ser operada, evitando, assim, choques elétricos. Outro ponto positivo do RFID é que por não necessitar de manter linha de visão entre a etiqueta e leitor, ele pode ler grande quantidade de produtos sem que precisem ser posicionados em frente à leitora, fazendo com que o tempo seja minimizado.

Além dessas vantagens, o RFID traz outros benefícios, tais como a automatização de diversos processos de uma empresa, como a montagem, o controle de qualidade, a diminuição do desperdício de maneira geral e o inventário desta empresa. Com isso há mais satisfação por parte dos clientes.

Como toda tecnologia há também as desvantagens na utilização do RFID, tais como, alto custo para se ter infraestrutura e para aquisição dos equipamentos, os quais podem ser facilmente interrompidos ou bloqueados, dependendo da energia aplicada ao sistema, bem como problemas de interferências com outras tecnologia ou matérias de aplicação.

1.2.10 Casos de Uso do RFID

O RFID apresenta diversas formas para sua aplicação, conforme pode ser exemplificado.

- a) Controle de acesso: o controle de acesso não tem aplicação específica, pois pode ser tanto de pessoas como de carros ou até de animais.
- b) Controle de produtos: neste caso, os produtos têm uma *tag* e nela carregam sua identidade. Com isso, o controle de estoque e de deslocamento pode ser feito de forma mais rápida, mais prática e mais eficiente.
- c) Identificação de pacientes: em hospitais, hoje, a identificação de pacientes é feita por uma pulseira que tem uma *tag* que contém informações como nome, idade, sexo e histórico do hospital. Assim, em vez de ter enormes prontuários, com apenas uma pulseira o médico pode saber rapidamente da situação de cada paciente. Mas acima de tudo evita erros médicos em razão de troca de prontuários, pois estas pulseiras não podem ser retiradas e caso sejam arrancadas podem ser estragadas.
- d) Identificação animal: muitas fazendas com grande quantidade de gado utilizam este tipo de tecnologia, pois assim podem ter um controle melhor, mais eficiente e mais fidedigno, evitando erros nos tratamentos e na criação dos animais. As etiquetas são geralmente em formato de brincos, pois ficam aplicadas nas orelhas dos animais.
- e) Sistemas antifurto: para que furtos sejam evitados, muitas lojas de roupas e de outros produtos têm uma pequena *tag*. Geralmente, nas roupas tem uma etiqueta que fica junto à etiqueta da loja. Assim, a pessoa ao passar pelo sensor nas portas das lojas aciona o alarme fazendo com que o furto seja descoberto. Algumas dessas etiquetas não têm nem mesmo identificação, elas apenas transmitem um sinal para o sensor quando este a aciona e, com isso, o alarme é tocado, o que serve para baratear o custo de utilização. Por esta razão é que, muitas vezes, quando a compra é efetivada, as roupas são passadas em equipamentos eletromagnéticos para desativar as etiquetas.
- f) Controle de frotas: no caso de frotas o RFID é bastante utilizado pelas empresas para que saibam onde realmente cada veículo de sua frota se encontra, e assim terem maior controle de suas despesas ou até mesmo do trabalho de seus funcionários. Cada carro carrega uma etiqueta que utiliza a frequência de RLTS,

para a identificação de pequeno espaço ou até mesmo em conjunto com o GPS para a localização na cidade, no estado ou no país.

- g) Identificação de bagagens em aeroportos: muitos aeroportos estão utilizando este tipo de identificação com o intuito de minimizar o extravio de bagagens e o envio de malas para destinos nos quais não deveriam ir. Isso porque é necessário que a identificação seja rápida e confiável, itens característicos de sistemas RFID.

Essa tecnologia foi escolhida por sua fácil aplicação e pela ótima divulgação e estudos hoje desenvolvidos. E por isso foi possível maior aprofundamento nessa tecnologia visando melhor adaptá-la ao projeto e às funções peculiares a ele. Além disso, sua relação com outras tecnologias, tais como a do código de barras, foi superior e mais confiável, conforme comparação feita no Quadro 1.2, pois o projeto busca agilizar o processo de entrada e de saída de funcionários, o que deverá ser automatizado. Para tanto, fez-se necessário uma tecnologia que atendesse estes requisitos, sem contar que o material a ser utilizado deveria ser de qualidade e ter durabilidade, tendo em vista que nesta aplicação empresarial o material é usado em grande quantidade e as etiquetas estariam nos mais diversos lugares.

2 REFERENCIAL TECNOLÓGICO

2.1 Linguagens de Programação

As linguagens de programação são basicamente conhecidas por interpretar o que os programadores querem e mandar o computador executar essas vontades, ou seja, as linguagens são os tradutores entre os programadores e os computadores.

Existem várias linguagens de programação, sendo que um programa pode ser constituído por um grupo de linguagens, ou seja, ele pode ser feito por partes e nessas partes por diferentes tipos de linguagens. Para o programa funcionar é necessário que esta programação seja compilada, e então entendida pelo computador, pois ao compilar um código fonte, o desenvolvedor não está fazendo nada além de transformar o que ele escreveu em linguagem que o computador entenda e, assim, executá-la.

Dependendo da linguagem empregada a compilação é feita de forma diferente. Por exemplo, na linguagem C o desenvolvedor pode fazer toda a sua programação e ao final dela fazer a compilação para que o computador execute o que o programador deseja e esta compilação fique armazenada na memória para que não seja necessário compilar todas as vezes que queira executar um programa, fazendo, assim, economia de tempo. No entanto, no caso do Java, que é outra linguagem, o código fonte só pode ser traduzido para a linguagem do computador na medida em que é executado, método este conhecido como interpretação, ou seja, à medida em que o código é feito e executado há uma interpretação do que está descrito, o que é passado para a linguagem de computador. A grande diferença entre estes dois métodos é que no caso da compilação, a execução de um programa é mais rápida, mas há a limitação na interatividade com o ambiente. No caso da interpretação, o programa, apesar de ser um pouco mais lento em sua execução, permite maior interatividade com o ambiente.

2.1.1 Linguagem PHP

O PHP foi um melhoramento de uma linguagem anterior a ele, conhecida como PHP/FI, que significa *Personal Home Page / Form Interpreter* (Página Inicial Pessoal / Intérprete de Formulário). O PHP/FI foi criado por Rasmus Lerdorf no ano de 1995, que fez esta linguagem inicial para que fosse possível acessar seu currículo *online*, sendo daí que vem o nome *Personal Home Page* ou Páginat Inicial Pessoal. No entanto, depois de

implementá-la ele descobriu que precisaria de mais funcionalidades e, para isso, escreveu um código muito maior em C, com mais funcionalidades, tais como, comunicação com banco de dados e possibilidade de desenvolver aplicativos para *web*. Após essa atualização, Lerdorf decidiu disponibilizar sua linguagem para que outras pessoas pudessem ajudá-lo, usar para suas aplicações e ajudar com a resolução de *bugs*. O PHP conhecido hoje é uma atualização da linguagem originalmente criada por Rasmus Lerdorf com variáveis Perl, interpretação automática de variáveis vindas de formulário e sintaxe embutida no HTML. O PHP de hoje foi criado por Andi Gutmans e Zeev Suraski em 1997, nada mais sendo que atualização e reescrita do PHP/FI, pois Gutmans e Suraski perceberam que poderiam melhorá-lo e utilizá-lo para desenvolverem sua aplicação de *e-commerce* como projeto de conclusão de curso da faculdade. Então, Andi, Rasmus e Zeev se juntaram para desenvolver uma linguagem a partir do zero, atualizando o PHP/FI que virou o PHP 3.0, colocado para testes pela comunidade. Hoje já existe o PHP 5.0 que é uma atualização do PHP 3.0 (Wikipedia – Linguagem PHP, 2010).

A programação da comunicação com o banco de dados, do atual projeto, como busca dos dados para relatório de ponto, cadastro dos funcionários e o processamento dos pontos destes funcionários, foi desenvolvida utilizando a linguagem PHP. Foi com esta linguagem que foi criada a parte estética do programa.

2.1.2 Linguagem C

A linguagem C teve início nos laboratórios Bell na década de 70 sendo criada a partir de uma linguagem anterior chamada de “B”, tendo, por isso, esse nome. Os primeiros a desenvolver esta linguagem foram Brian Kernigham e Dennis M. Ritchie.

O “C” é uma linguagem que pode ter diversos tipos de aplicação, como processadores de texto, sistemas operacionais e programas de comunicação. É considerada de nível médio, por ser muito próxima à linguagem humana e, com isso, ter uma gama de instruções de alto nível. Ao mesmo tempo em que tem sintaxes muito parecidas com as sintaxes humanas, ela tem, também, instruções de baixo nível como a manipulação de *bits*, *bytes* e endereços de memórias. Isso faz com que ela tenha ampla diversidade de aplicação (WIKILIVROS – Linguagem C, 2010).

No projeto, a linguagem C foi utilizada para a comunicação do computador com o circuito via porta paralela. Com esta programação é possível que o sinal seja enviado para o pino certo e que o comando certo seja executado.

2.2 Banco de Dados MySQL

O MySQL é rápido e estável, sendo esta a chave do seu sucesso. Em fevereiro de 2002, a *eWeek* (www.eweek.com) fez um estudo de performance com os maiores sistemas de banco de dados, incluindo Oracle, Microsoft SQL Server, DB2 e MySQL, entre outros. O estudo avaliou o MySQL e o Oracle como tendo os melhores desempenhos em geral. Cabe ressaltar que o estudo foi realizado utilizando a versão Alfa do MySQL 4.0, no entanto, a versão 4.1 é ainda mais rápida (Welling e Thomson, 2004).

O MySQL está disponível através de um esquema de licenciamento duplo, tanto como um *software* livre quanto como um *software* comercial. Todo o *software* MySQL é disponibilizado sob licenciamento *GNU General Public License* (GPL) (Welling e Thomson, 2004).

O MySQL suporta a maioria dos recursos considerados importantes pelos usuários e administradores de bancos de dados, como, por exemplo, transações, *lock* de registros, chaves estrangeiras, subconsultas e pesquisa *full-text*. Na versão 5.0 o *stored procedures* é acrescentado nessa lista (Welling e Thomson, 2004).

O MySQL é um *software* robusto e, ao mesmo tempo, flexível. As qualidades do MySQL justificam sua escolha por empresas com demandas gigantescas de gerenciamento de dados como Yahoo!, Finance, Slashdot e o U.S. Census Bureau (Welling e Thomson, 2004).

Devido à facilidade de instalação e uso, o MySQL é uma ótima ferramenta de aprendizado a respeito dos bancos de dados. Somam-se a isso os baixos requisitos de recursos de *hardware*, como disco rígido e memória (Welling e Thomson, 2004).

O MySQL é um Sistema de Gerenciamento de Banco de Dados (ou simplesmente SGBD), e, para seu processamento, utiliza o SQL (*Structured Query Language*, ou Linguagem de Consulta Estruturada).

O banco de dados foi usado para armazenamento dos dados dos funcionários e da identificação dos cartões utilizados por eles para que tivessem a entrada permitida.

2.3 Porta Paralela

A porta paralela é uma interface de comunicação entre um computador e um periférico. A IBM (*International Business Machines*) foi uma das primeiras empresas a desenvolver a porta paralela como maneira de ligar a impressora ao computador. Atualmente

a porta paralela não é usada somente para impressora. Pode-se desenvolver um circuito eletrônico e acoplá-lo a essa porta e, através de um programa específico, enviar-lhe sinais digitais para controlá-lo. Na comunicação em paralelo, grupos de *bits* são transferidos simultaneamente, em geral, *byte a byte*, através de diversas linhas condutoras dos sinais. Desta forma, como vários *bits* são transmitidos simultaneamente a cada ciclo, a taxa de transferência de dados (*throughput*) é alta. Há dois tipos de modelos de porta: paralela à unidirecional e as bidirecionais (ROGERCOM – Porta Paralela, 2010).

A unidirecional SPP (*Standard Parallel Port*) pode chegar a uma taxa de transmissão de dados a 150KB/s. Ela se comunica com a CPU (*Central Processing Unit*) utilizando um barramento de dados de 8 *bits*. Para a transmissão de dados entre periféricos são usado 4 *bits* por vez. Ela apresenta apenas 3 grupos de registros: o registro de dados, o registro de estado e o registro de controle (ROGERCOM, 2010).

Existem dois tipos de bidirecionais: EPP (*Enhanced Parallel Port*) e a ECP (*Enhanced Capabilities Port*).

A porta avançada EPP chega a atingir uma taxa de transferência de 2 MB/s, mas para atingir essa velocidade é necessário um cabo especial. Ela então se comunica com a CPU utilizando um barramento de dados de 32 *bits*. Para a transmissão de dados entre periféricos são usado 8 *bits* por vez (ROGERCOM – Porta Paralela, 2010).

A porta avançada ECP tem as mesmas características que a EPP, porém, utiliza DMA (acesso direto à memória), sem a necessidade do uso do processador, para a transferência de dados. Utiliza, também, um buffer FIFO (*First In First Out*) de 16 bytes (ROGERCOM – Porta Paralela, 2010).

A porta paralela é para a comunicação entre o computador, utilizando a programação feita para o funcionamento do sistema e o circuito para o acendimento dos LEDs e da abertura da fechadura.

2.3.1 Endereçamento de Porta Paralela

Um computador geralmente tem uma porta paralela fisicamente instalada, denominada de LPT1, mas pode criar várias portas denominadas LPT1, LPT2 e assim sucessivamente. A seguir, o Quadro 2.1, que demonstra o endereço lógico de cada porta.

Quadro 2.1 – Endereços de Porta Paralela

Nome da Porta	Endereço de Memória	Endereço de Porta		Descrição
LPT1	0000:0408	378 hexadecimal	888 decimal	Endereço Base
LPT2	0000:040A	278 hexadecimal	632 decimal	Endereço Base

Fonte: ROGERCOM, 2010

Pode-se observar que na tabela, na coluna “Descrição” aparece “Endereço Base” porque o 378 hexadecimal é o endereço base da porta LPT1 e o 278 hexadecimal para o LPT2. No entanto, dependendo do tipo de dado enviado ou recebido há um endereço diferente. Por exemplo:

- 378h – envio de um *byte* de dados pela porta;
- 378+1h – recebe um valor através da porta;
- 378+2h – envia dados.

E, assim, para a porta LPT2, caso esta exista, seguindo sempre seu endereço base.

2.3.2 Conector DB25

O DB25 é um conector que fica na parte traseira do gabinete do computador e é através dele que o cabo paralelo se conecta ao computador para poder enviar e receber dados. No DB25 um pino está em nível lógico 0 quando a tensão elétrica dele está entre 0 a 0,4v. Um pino encontra-se em nível lógico 1 quando a tensão elétrica nele está acima de 3.1 e até 5v (ROGERCOM - Porta Paralela, 2010). Nas Figuras 2.1 e 2.2, as portas paralelas são ilustradas.

DB25 que fica atrás do Micro.

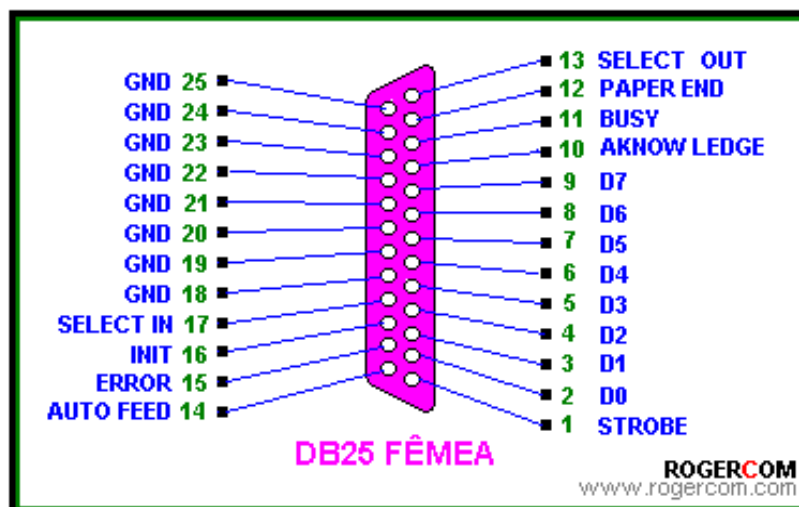


Figura 2.1 – Porta Paralela do Computador
Fonte: ROGERCOM.

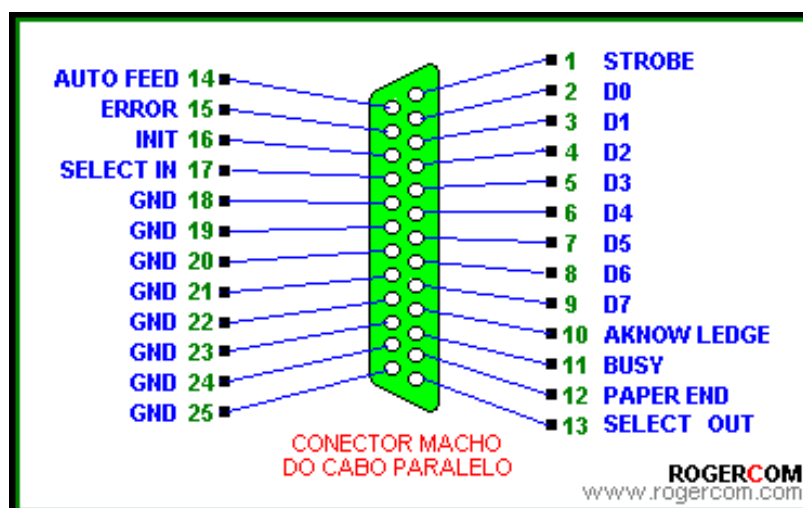


Figura 2.2 – Conector Macho do Cabo Paralelo
Fonte: ROGERCOM.

2.3.2.1 Pinagem

A Figura 2.3 mostra uma porta paralela e sua pinagem, sendo 17 pinos utilizados para comunicação e 8 pinos que são para a ligação terra.

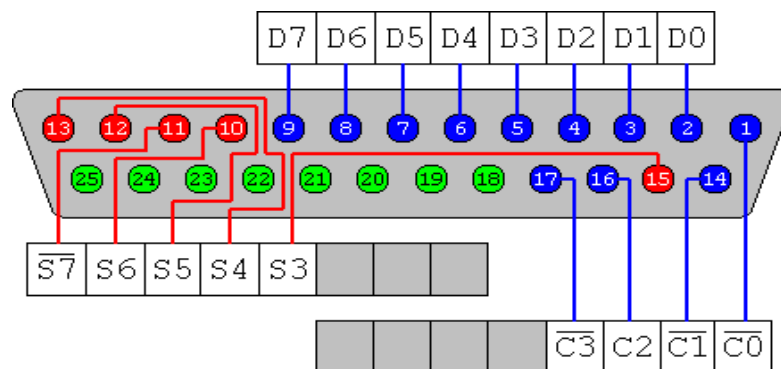


Figura 2.3 – Pinagem da Porta Paralela

Fonte: Tugatronica.

- Os pinos 2 a 9 são controlados pelo registro DADOS, tendo como função enviar dados através da porta paralela. As implementações novas permitem comunicação bidirecional por meio destas linhas.
- Os pinos 10, 11, 12, 13 e 15, são usados para troca de mensagens, entre dispositivos para o PC. O registro que os controla é o registro STATUS.
- Os pinos 1, 14, 16 e 17 são usados para a interface, controle e troca de mensagens do PC para o dispositivo. São controlados pelo registro CONTROLO.
- Os pinos 18 a 25 são terra ou GND como comumente conhecidos.

Existem alguns pinos que podem trabalhar com o sinal invertido, ou seja, quando é preciso ativar o pino é necessário enviar um sinal lógico “0”, e para desativar é necessário enviar um sinal “1”. A Figura 2.4 mostra as funções detalhadas de cada pino e se necessita inverter o sinal enviado.

<u>Pino - Função</u>	<u>Pino - Função</u>
1 - Store	14 - Auto Feed
2 - S0	15 - Error
3 - S1	16 - Init
4 - S2	17 - Slct In
5 - S3	18 - GND
6 - S4	19 - GND
7 - S5	20 - GND
8 - S6	21 - GND
9 - S7	22 - GND
10 - Acknowledge	23 - GND
11 - Busy	24 - GND
12 - Paper End	25 - GND
13 - Slct Out	

Figura 2.4 – Descrição da pinagem da porta Paralela
 Fonte: <http://juliusesar.vilabol.uol.com.br>.

2.4 Porta Serial

Considerada uma das conexões externas mais básicas para um computador, a porta serial é parte integrante da maioria dos computadores há mais de 20 anos. Embora muitos sistemas novos tenham abolido a porta serial completamente, colocando em seu lugar conexões USB, a maioria dos *modems* ainda utiliza a porta serial, assim como algumas impressoras, PDAs e câmeras digitais. Poucos computadores possuem mais de duas portas seriais com conectores DB9 (HowStuffWorks – Porta Serial, 2010).

A tensão nos pinos do conector pode estar em dois estados: ligada ou desligada. Ligado (valor binário "1") significa que o pino está transmitindo um sinal entre -3 e -25 volts, enquanto desligado (valor binário "0") significa que ele está transmitindo um sinal entre +3 e +25 volts (HowStuffWorks – Porta Serial, 2010).

No projeto, a porta serial leva a leitora a comunicar-se com o computador e assim os dados lidos são recebidos e comparados com os do banco de dados para que a abertura da fechadura seja autorizada ou não.

2.5 Fechadura Eletromagnética

A fechadura eletromagnética utiliza a energia elétrica e um pequeno campo magnético para funcionar, ou seja, ao ser carregada com eletricidade a fechadura gera um campo magnético que faz com que a trava se mova para a posição de aberta e, com isso, abre a fechadura e, conseqüentemente, a porta. Ao fechar a porta, a trava da fechadura volta ao normal, e assim que não houver mais eletricidade, o campo magnético formado se desfaz e a trava volta ao seu lugar de origem, fazendo com que a porta fique travada. Este tipo de fechadura é muito útil para travar portões externos de casas, mas existem várias outras fechaduras que podem ser colocadas em portas de vidro, ou em portas normais, sendo estas mais discretas. A fechadura da Figura 2.5 é para uso externo e é utilizada no projeto proposto.

A fechadura tem a função, no projeto, de fazer a porta permanecer trancada caso a entrada não seja permitida. Por este motivo essa fechadura foi escolhida, para que somente se necessário ela seja aberta de forma automática e sem interferência humana.



Fonte: AGL Fechaduras.

Figura 2.5 – Fechadura Eletromagnética

3 IDENTIFICAÇÃO E PONTO DE PRESENÇA POR ETIQUETA RFID

Neste capítulo são apresentados a implementação, os testes e os resultados obtidos durante a criação do projeto, bem como a simulação do seu funcionamento.

3.1 Ordem de Criação e Planejamento do Projeto

O projeto foi desenvolvido em quatro etapas, sendo a primeira para as pesquisas bibliográficas e o planejamento da parte prática, ou seja do protótipo; a segunda para a exposição teórica da monografia, não incluindo, portanto, as partes que não envolvessem os testes e o desenvolvimento prático do projeto; a terceira para a criação da parte prática do projeto, tais como a criação e a confecção do circuito, o desenvolvimento da programação e o banco de dados, a integração de sistemas e testes finais. A quarta e última etapa foi destinada à redação final da monografia, tais como conclusão, problemas encontrados, testes e ajustes finais. Para mais clareza em relação ao desenvolvimento deste trabalho ilustra-se na Figura 3.1 as etapas de realização do projeto.

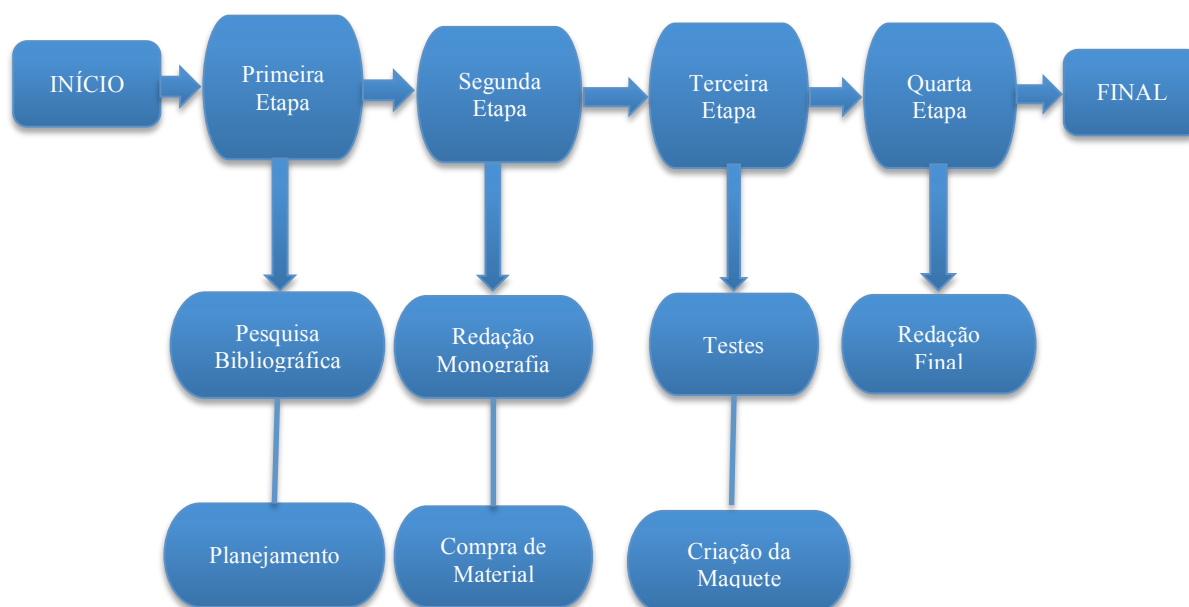


Figura 3.1 – Ordem de planejamento do Projeto

3.1.1 Estrutura Geral do Projeto

Este projeto consiste em representar uma automatização no controle de acesso de pessoas em determinadas áreas de uma empresa ou instituição de ensino e em registrar o horário de entrada e de saída destas pessoas. O objetivo é permitir que apenas quem é autorizado possa entrar em determinadas áreas, além de automatizar a marcação de entrada e de saída nestas dependências, minimizando erros que possam vir com estas marcações, uma vez que estes horários estarão armazenados dentro de um banco de dados. O projeto alicerçou-se nos seguintes componentes eletrônicos:

- Leitora RFID AP-09 (Acura Technologies);
- Cartões RFID (Acura Technologies);
- Circuito Eletrônico para comando de abertura da fechadura (Desenvolvimento próprio);
- Fechadura eletromagnética.

Esses dispositivos são controlados por uma integração de sistemas que envolvem as linguagens PHP, para o controle de cadastros, elaboração de relatórios e controle do banco de dados que foi elaborado utilizando o MySQL, C para o envio do comando para o circuito que controla os LEDs vermelho e verde e o acionamento da fechadura. Para a integração do sistema com o *hardware* do projeto foram utilizados os seguintes métodos:

- Cabo paralelo – para a comunicação entre o *software* e o circuito eletrônico;
- Cabo TTL – para a comunicação entre a leitora e o adaptador para a porta serial;
- Adaptador TTL para serial – para o recebimento do sinal TTL da leitora para o envio para a porta serial;
- Cabo Paralelo – para a comunicação entre o adaptador e a porta serial do computador.

3.1.2 Funcionamento Básico do Projeto

A Figura 3.2 ilustra como o projeto funcionará, exemplificando as verificações e os comandos de execução, bem como sua ordem de execução.

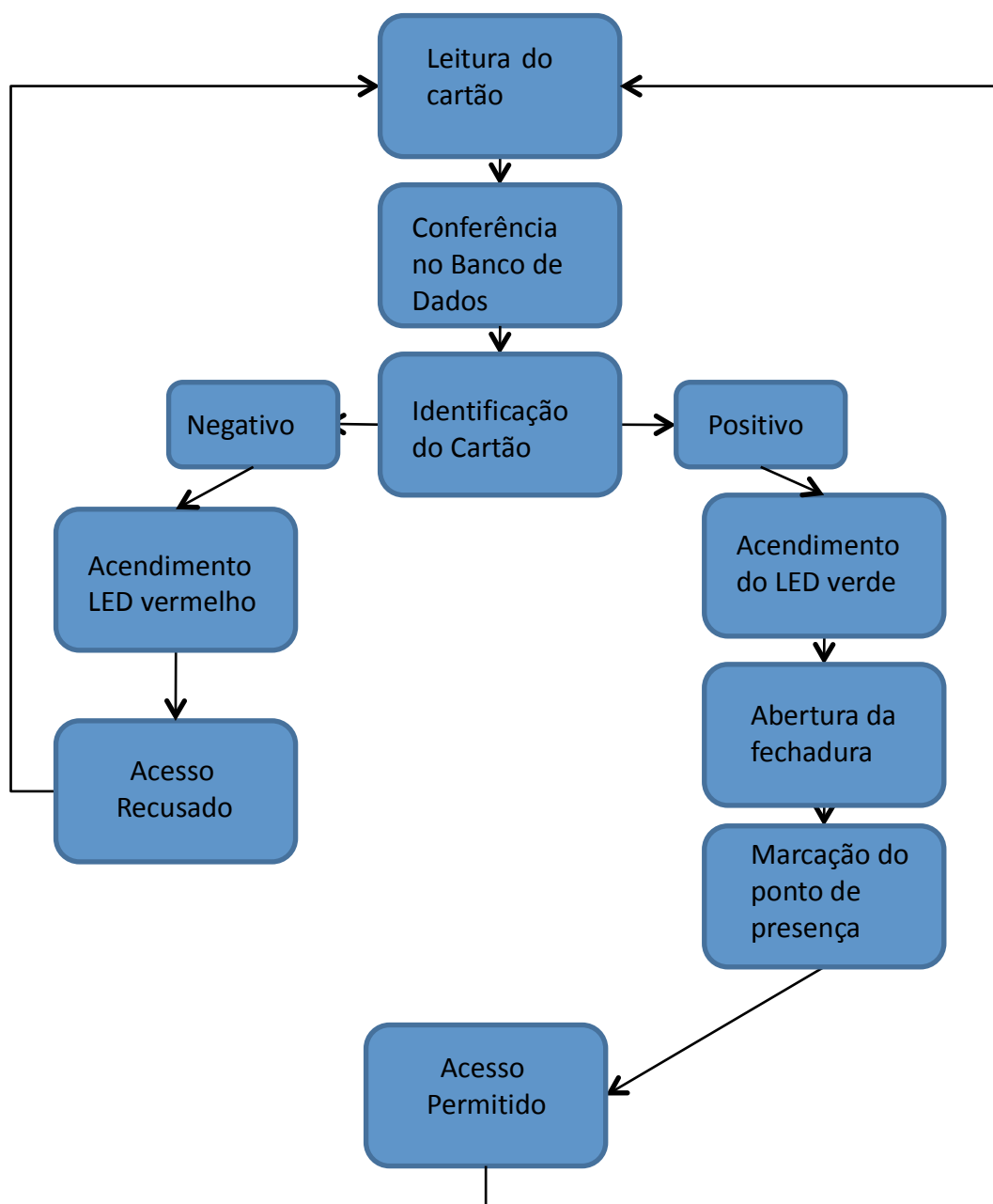


Figura 3.2 – Fluxograma do funcionamento do Projeto

Observando-se a Figura 3.2 é possível compreender o funcionamento geral do projeto. Uma pessoa aproxima o cartão da leitora que recebe o sinal e o envia para o computador onde o sistema irá verificar se o ID do cartão se encontra no banco de dados. Quando não existe o ID no banco de dados o sistema envia um comando para o circuito eletrônico que, por sua vez, acende um LED vermelho informando ao usuário que o acesso foi negado devido à não identificação do ID. Se a identificação for positiva, o sistema enviará o comando para o circuito que libera o acesso, ou seja, que destrava a fechadura, e acenderá o LED da cor verde, assim marcando o horário da entrada ou da saída da pessoa.

3.2 Dispositivos Eletrônicos do Projeto

O projeto compõe-se de alguns dispositivos eletrônicos, os quais são:

- Leitora AP-09;
- Circuito eletrônico;
- Fechadura eletromagnética.

O circuito eletrônico foi confeccionado pelo autor do projeto que utilizou para esta criação o *software* Proteus versão 6.0. A comunicação entre o circuito e o computador foi feita através de um cabo paralelo.

A figura 3.3 ilustra o circuito criado.



Figura 3.3 – Circuito Eletrônico

A Figura 3.4 ilustra como foi projetado o circuito e suas ligações entre a porta paralela e os LEDs e o relé que envia o sinal para a fechadura; e a figura 3.5 mostra a forma em que as conexões (chamadas trilhas) foram feitas também a partir do *software* Proteus. No Quadro 3.1 observa-se as características de cada pino da porta paralela.

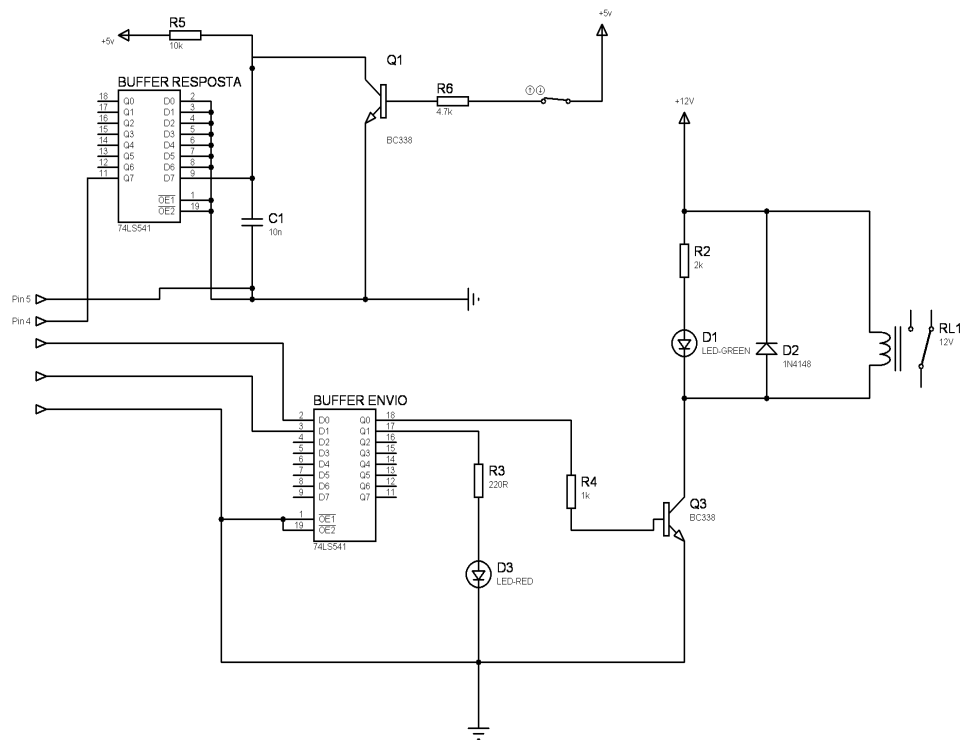


Figura 3.4 – Mapa do circuito e componentes

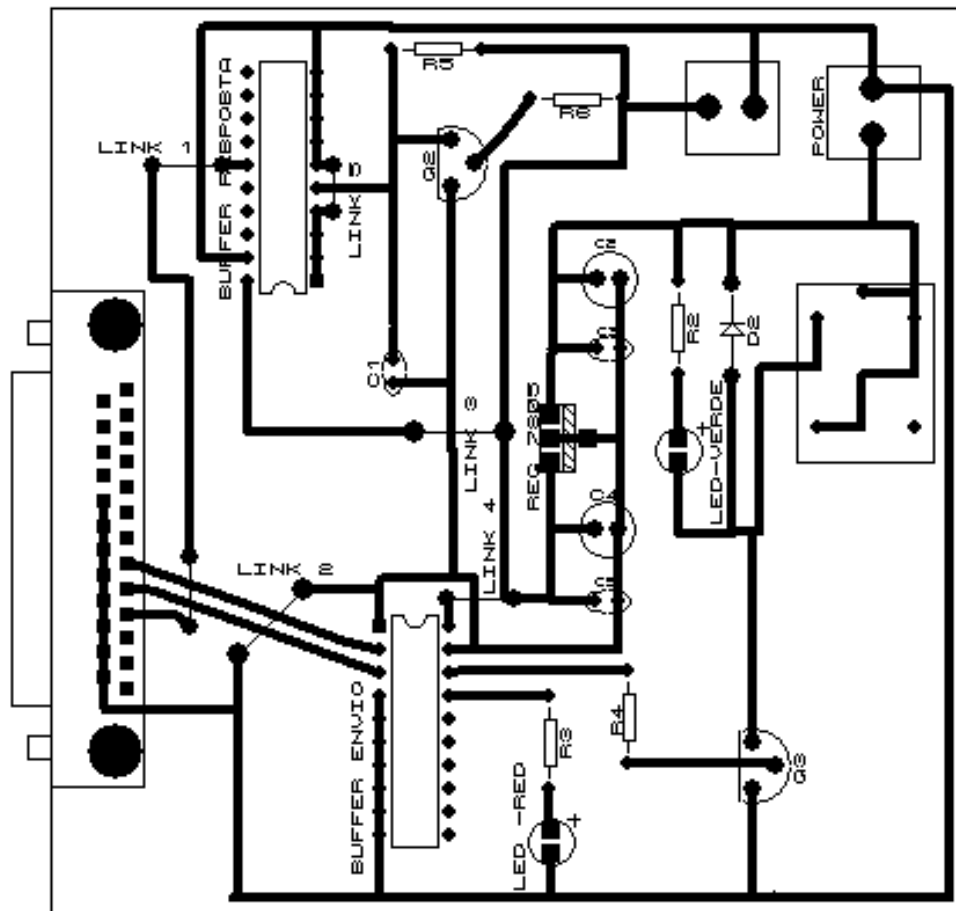


Figura 3.5 – Mapa de trilhas do circuito

Quadro 3.1 – Descrição da utilização dos pinos da porta paralela

Pinos referentes à porta paralela	Observação
Pinos 1 a 7	Pinos disponíveis por não terem sido utilizados
Pino 8	Envio de sinal para o LED vermelho
Pino 9	Envio de sinal para o LED verde e relé
Pino 10	Recebe sinal de equipamento externo
Pino 11 a 17	Pinos disponíveis por não terem sido utilizados
Pinos 18 a 25	Pinos Terra.

3.2.1 Especificações dos dispositivos utilizados no Projeto

Neste subitem, além das especificações de cada dispositivo utilizado há também a especificação do circuito usado no projeto.

3.2.1.1 Leitora RFID AP-09 Acura Technologies

A leitora (Figura 3.6) que serve para a conferência dos cartões RFID foi a AP-09 fabricada pela *Acura Technologies*. Esta leitora é fabricada com o intuito de desenvolvimento de sistemas, e, por isso, apresenta forma mais rústica e sem muitos acabamentos. Ela transfere os dados recebidos através do protocolo TTL, sendo, por isso, necessário um adaptador TTL – porta serial, para que os dados possam chegar ao computador. Para seu funcionamento é preciso uma tensão de 5v a 13,5v. Sua distância de leitura é de 9cm à 12v, mas como o projeto é apenas um protótipo, esta distância é suficiente. A distância do cabo é de, no máximo, 15 metros para utilização do protocolo serial; a frequência utilizada é de 125 kHz; possui 3 protocolos de comunicação os quais são:

- Wiegand (26 bits);
- Magstripe (ABA TK2);
- Serial ASCII (TTL RS-232).



Figura 3.6 – Leitora RFID AP-09

3.2.1.2 Circuito Eletrônico

O circuito eletrônico deste projeto foi de fabricação do próprio autor, que utilizou o *software* Proteus para auxiliá-lo na confecção. Este circuito consiste em receber o sinal da porta paralela e endereçá-lo ao componente desejado, ou seja, quando o circuito recebe o sinal ele mesmo acende o LED verde e aciona o relé para a abertura da fechadura em caso de acesso permitido; e em caso de acesso negado ele acende o LED vermelho. Permite, também, receber o sinal externo, caso necessário, e enviá-lo para o computador utilizando a porta paralela.

Os componentes para a criação do circuito foram:

- Buffer 74LS541N – este *buffer* é empregado como um *driver* de endereço de memória, geralmente para microcontroladores. Neste projeto ele serviu para destinar o sinal para os LEDs e para o relé, sem erros.
- LEDs – são diodos emissores de luz e podem estar tanto em relógios quanto em painéis de *outdoors* ou até mesmo em televisores. Foi usado no projeto para sinalizar a permissão ou o bloqueio do acesso (HowStuffWorks - LED, 2010).
- Regulador de tensão LM7805 – usa-se bastante este componente para regular a tensão de um circuito, ou seja, para fazer entrar uma tensão de 12v e sair uma tensão de 5v.
- Capacitores – tem funcionamento semelhante ao da pilha que armazena a energia recebida. Ele envia a energia na medida que recebe tensão evitando,

assim, um envio abrupto de energia para um componente mais sensível (HowStuffWorks - Capacitores, 2010).

- Resistores – os resistores nada mais são do que uma resistência à passagem de energia, podendo ter resistência fixa ou variável.
- Relé – o relé funciona criando um campo magnético pela bobina que possui internamente, o qual uma vez criado atrai o contato para abrir ou fechar o circuito. No circuito do projeto serve para enviar a tensão que abre a fechadura.
- Transistores – os transistores servem para amplificar ou interromper sinais elétricos. Basicamente, um transistor possui uma combinação de dois diodos de junção PN, tendo uma das junções polarizada diretamente e a outra de forma inversa. Assim, pode-se ter duas formas de união: uma através do material P, para que seja possível produzir um transistor NPN e outra através do material N, para a produção de um transistor PNP (HowStuffWorks - Transistores, 2010).

3.2.1.3 Fechadura Eletromagnética

A fechadura eletromagnética gera um campo magnético no seu interior para mover a trava nela existente, de forma que seja possível abrir ou travar a porta. Esse campo é gerado à medida que possui ou não tensão na fechadura. Quando há tensão a trava é deslocada para trás, pelo campo eletromagnético, fazendo com que a fechadura se abra. Não havendo esta tensão o campo é finalizado, a trava volta para o seu lugar de origem; ao fechar a porta o trinco é travado e a porta é impossibilitada de ser aberta. Nas figuras 3.7 e 3.8 podem ser vistos os estados de abertura e de fechamento da trava.

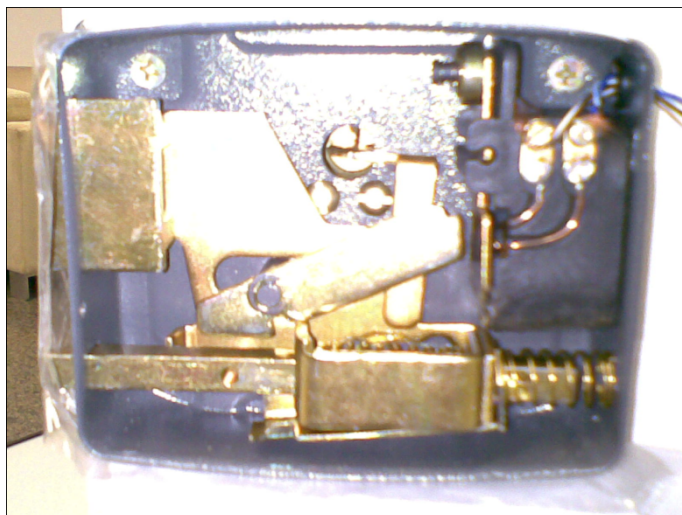


Figura 3.7 – Fechadura aberta

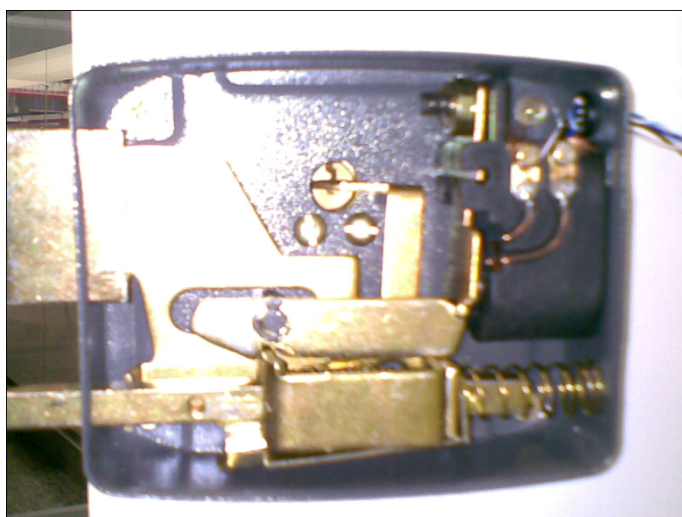


Figura 3.8 – Fechadura fechada

3.3 Software

Na integração entre os dispositivos do projeto usou-se para desenvolver a programação, o *software* de desenvolvimento DEV C++, para o desenvolvimento dos comandos em C; o *software Intype* para o desenvolvimento em PHP para a comunicação entre a *interface* e o banco de dados. No desenvolvimento do banco de dados foi utilizado o MySQL Server 5.1 e o MySQL Workbench 5.2.29. Na integração entre o computador e a leitora foi usado o software COMM-Logger da Willies Computer Software, visando à geração de um *log* em extensão txt para a conferência com o banco de dados.

3.3.1 Principais Funções Utilizadas na Programação do Projeto

Nesta etapa são demonstradas as funções principais que foram aplicadas no desenvolvimento do programa do projeto.

3.3.1.1 Porta Paralela DB25

No Windows® XP e nos que fazem uso da plataforma NT a comunicação direta com a porta paralela não é possível, sendo então necessária a criação de uma DLL (*Dynamic Link Library*) que viabilize a comunicação. No projeto foi usada a DLL inpout32 que realiza a comunicação entre o sistema e a porta paralela do computador, sendo que esta foi adquirida por meio do *site* www.rogercom.com. Essa DLL tem uma função *main*, que a carrega para a comunicação com a porta paralela e que também recebe o endereço da porta por meio da qual se deseja receber um dado e também o dado que deseja ser enviado, no caso do projeto um *byte* de sinal.

3.3.1.2 RFID

Na comunicação com a porta serial fez-se uso de um programa COMM-Logger da Willies Computer Software, para criar um *log* em extensão txt e assim ser captado pelo programa do banco e conferido se o cartão pertence ao banco de dados ou não.

3.3.1.3 Banco de Dados

Para a realização do projeto buscou-se um Banco de Dados MySQL usando a versão do servidor 5.1. O comando para a conexão com o banco de dados foi desenvolvido em PHP:

```
$conexao = mysql_connect("localhost", "root", "")
```

As *query*sempregadas no banco de dados foram:

Para a criação do relatório do ponto - SELECT FC.func_nome as nome, FC.func_cargo as cargo, FC.func_matrícula as matrícula, FC.func_id as func_id, PT.ponto_hora as marcação, PT.ponto_tipo as tipo;

Para a marcação do ponto de entrada e saída da pessoa - INSERT into tb_ponto set ponto_hora = now(), func_tag = "" . \$func_tag . "", ponto_tipo = "" . \$ponto_tipo . ""

Para a exibição das pessoas cadastradas - SELECT func_id, func_nome, func_matrícula, func_tag, func_cargo from tb_funcionários”;

Para a atualização dos dados dos funcionários - UPDATE tb_funcionários set func_nome = "" . \$func_nome . "", func_matrícula = "" . \$func_matrícula . "", func_tag = "" . \$func_tag . "", func_cargo = "" . \$func_cargo . "" where func_id = "" . \$func_id . "" ”;

Para inclusão de novo funcionário - INSERT into tb_funcionários set func_nome = "" . \$func_nome . "", func_matrícula = "" . \$func_matrícula . "", func_tag = "" . \$func_tag . "", func_cargo = "" . \$func_cargo . "" ”;

Para a exclusão de novo funcionário - DELETE from tb_funcionários where func_id = "" . \$func_id . "" ”;

A complementação do código é encontrada no Anexo desta monografia, sendo este código fonte do sistema criado em PHP para processamento do ponto e cadastro de funcionários. Ele serve de comunicação com o banco de dados e também encontra-se em anexo o código fonte em C para a comunicação com a porta paralela e o circuito.

3.4 Testes e Resultados

No início do projeto tinha sido planejado que somente a porta paralela seria utilizada, mas ao adquirir a leitora RFID foi preciso incluir a porta serial, uma vez que a leitora usaria esta porta para enviar seus dados recolhidos. Como o controle da porta paralela era crucial, o teste teve início pela execução do controle e envio de dados para a porta paralela. As Figuras 3.9 e 3.10 ilustram os primeiros testes do circuito criado para o controle dos LEDs e do relé.

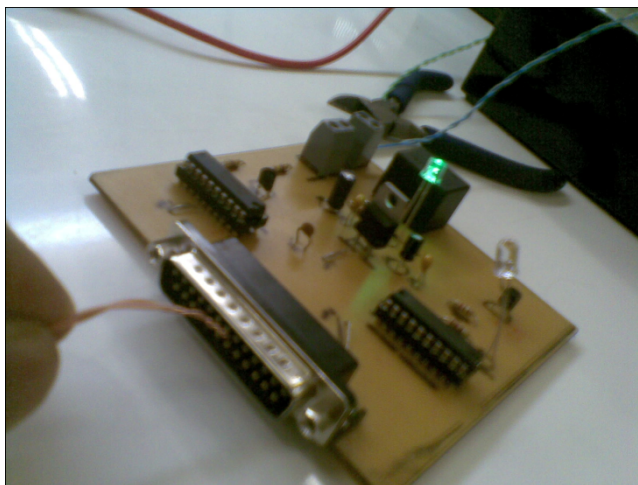


Figura 3.9 – Teste circuito acesso permitido

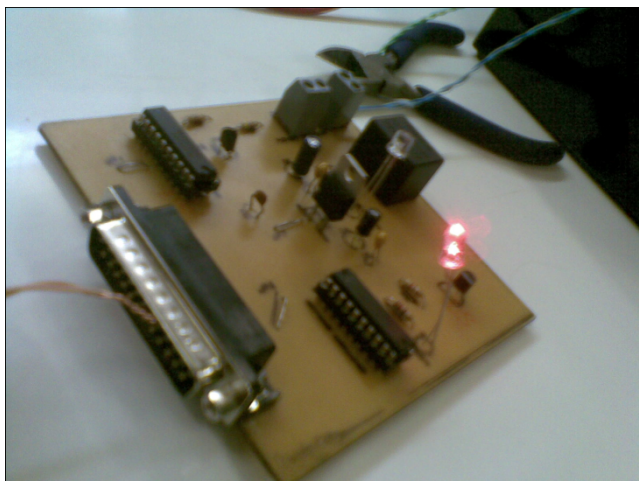


Figura 3.10 – Teste circuito acesso negado

O teste seguinte foi o de abertura da fechadura fazendo uso do próprio circuito e o início da utilização da porta paralela. A Figura 3.11 ilustra o teste realizado entre o circuito e o protótipo da porta.

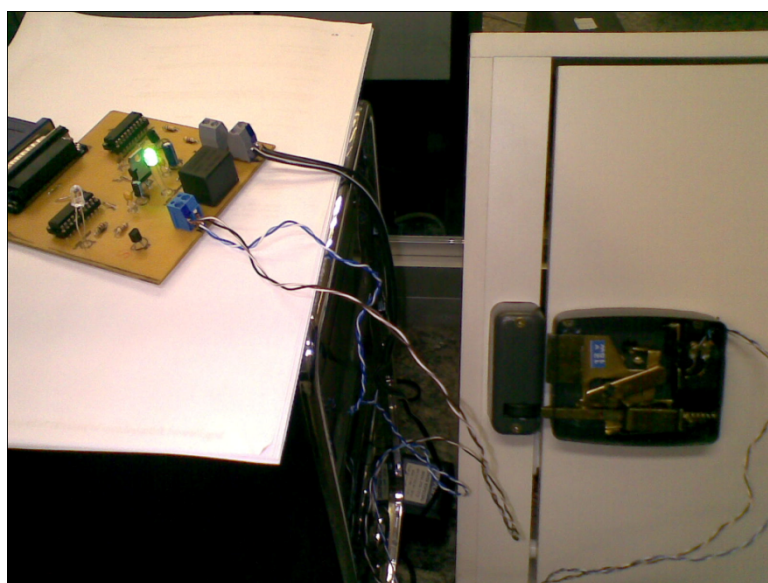


Figura 3.11 – Teste abertura porta com circuito

O passo seguinte foi a criação do teste com o banco de dados para verificar o cadastramento de funcionários e a exibição de relatórios. A Figura 3.12 mostra o *menu* do programa, que contém as opções de cadastro de funcionários, para incluir novos, alterar ou excluir antigos funcionários; a opção de relatório, que exibe os pontos marcados por nome de funcionário ou por data e também o item bater ponto, que serve para inicializar o processamento dos pontos.

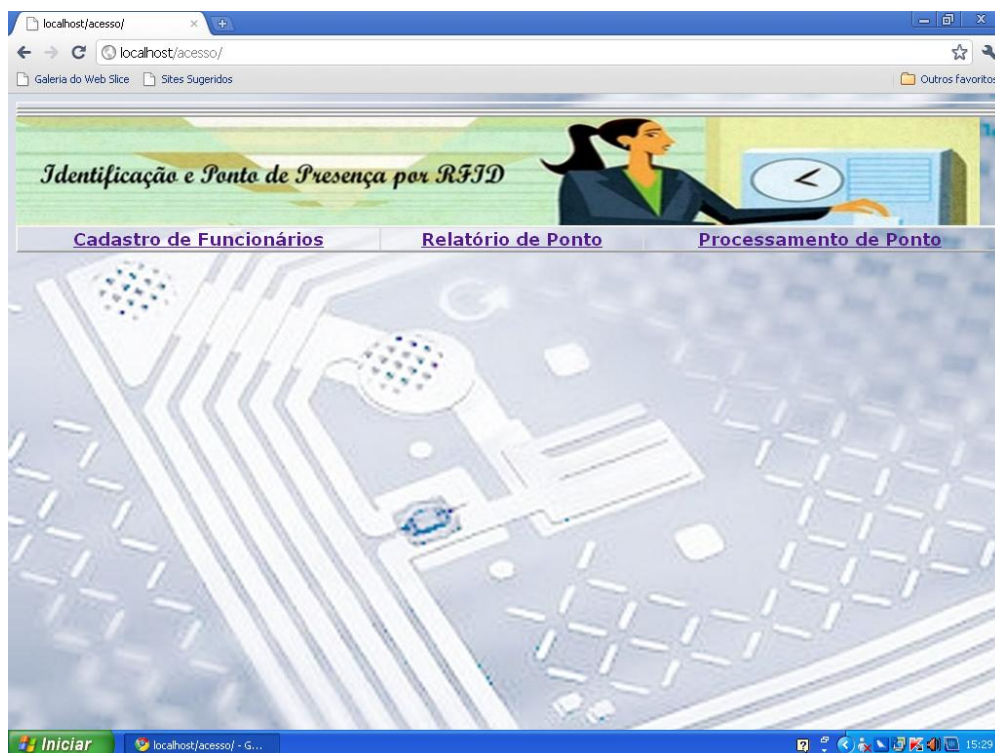


Figura 3.12 – Tela menu

A Figura 3.13 ilustra a opção de criar novos cadastros de funcionários.

Incluir Funcionário

Cadastro de Funcionários

Nome	Matrícula	Crachá	Cargo	Operação
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="button" value="Incluir"/>

Funcionários

Nome	Matrícula	Crachá	Cargo	Operação
Rudolf	1	3f6667e666f953e6a9f3a6cf4f00	Diretor	<input type="button" value="Editar"/> <input type="button" value="Excluir"/>
Paula	2	3fa6e6e64cb9b3c9d326a6cf4f00	Diretora	<input type="button" value="Editar"/> <input type="button" value="Excluir"/>
Marcelo	3	3f6667e666f973a666a629d6f200	Diretor	<input type="button" value="Editar"/> <input type="button" value="Excluir"/>
Cândida	4	202	Diretora	<input type="button" value="Editar"/> <input type="button" value="Excluir"/>
Francisco	5	350	Diretor	<input type="button" value="Editar"/> <input type="button" value="Excluir"/>
Flávia	6	265	Diretora	<input type="button" value="Editar"/> <input type="button" value="Excluir"/>

Figura 3.13 – Tela inclusão de funcionários

A Figura 3.14 mostra a alteração dos dados de um funcionário anteriormente cadastrado.

Nome	Matrícula	Crachá	Cargo	Operação
Rudolf	1	3f6667e666f953e6a9f3a6cf4f00	Diretor	Alterar

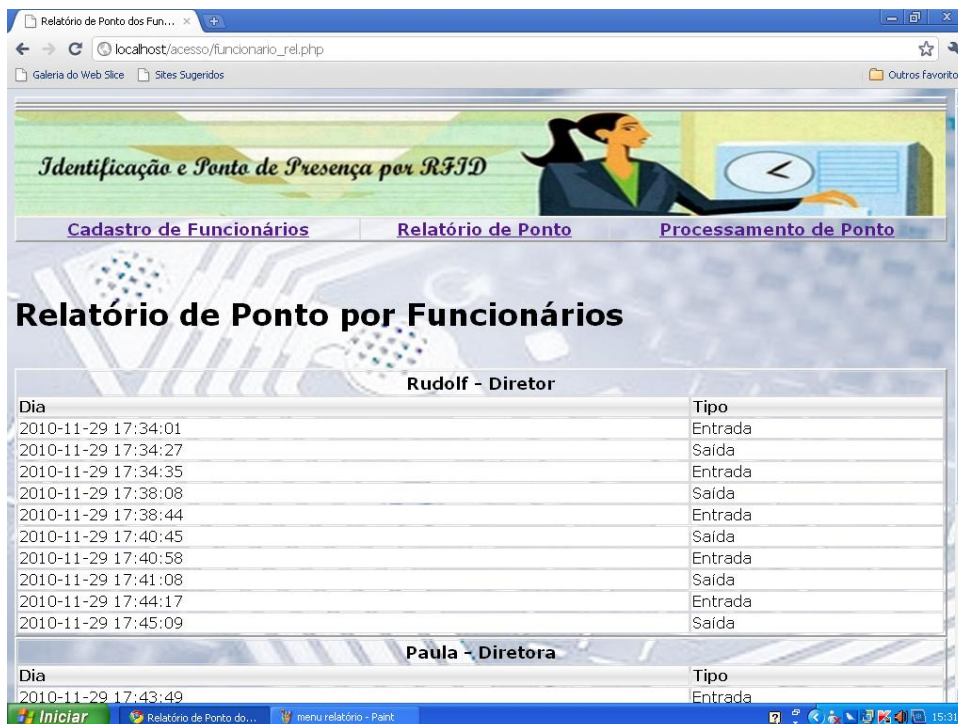
Nome	Matrícula	Crachá	Cargo	Operação
Rudolf	1	3f6667e666f953e6a9f3a6cf4f00	Diretor	Editar Excluir
Paula	2	3fa6e6e64db9b3c9d326a6cf4f00	Diretora	Editar Excluir
Marcelo	3	3f6667e666f973a666a629d6f200	Diretor	Editar Excluir
Cândida	4	202	Diretora	Editar Excluir
Francisco	5	350	Diretor	Editar Excluir
Flávia	6	265	Diretora	Editar Excluir

Figura 3.14 – Tela alteração de cadastro

A Figura 3.15 ilustra o *menu* para a exibição do relatório do ponto dos funcionários, podendo ser exibido tanto por nome de funcionário como por data e horário de entrada.

Figura 3.15 – Tela menu de relatório

Na Figura 3.16 é possível verificar a exibição do relatório por nome de funcionário.



Rudolf - Diretor	
Dia	Tipo
2010-11-29 17:34:01	Entrada
2010-11-29 17:34:27	Saída
2010-11-29 17:34:35	Entrada
2010-11-29 17:38:08	Saída
2010-11-29 17:38:44	Entrada
2010-11-29 17:40:45	Saída
2010-11-29 17:40:58	Entrada
2010-11-29 17:41:08	Saída
2010-11-29 17:44:17	Entrada
2010-11-29 17:45:09	Saída

Paula - Diretora	
Dia	Tipo
2010-11-29 17:43:49	Entrada

Figura 3.16 – Folha de ponto por nome

Na Figura 3.17, é possível identificar a exibição do relatório da folha de ponto dividido por data.



2010-11-29		
Hora	Funcionario	Tipo
17:32:43	Marcelo	Entrada
17:33:02	Marcelo	Saída
17:34:01	Rudolf	Entrada
17:34:27	Rudolf	Saída
17:34:35	Rudolf	Entrada
17:34:48	Marcelo	Entrada
17:34:55	Marcelo	Saída
17:38:08	Rudolf	Saída
17:38:13	Marcelo	Entrada
17:38:44	Rudolf	Entrada
17:40:45	Rudolf	Saída
17:40:58	Rudolf	Entrada
17:41:08	Rudolf	Saída

Figura 3.17 – Folha de ponto por data

O último teste foi realizado com o RFID identificando os cartões e marcando o ponto. Na Figura 3.18 ilustra-se esta etapa.



Figura 3.18 – Cartão identificado e ponto marcado

3.5 Simulação

A simulação do projeto foi realizada em uma maquete, tendo como equipamentos aqueles descritos como parte do projeto, tais como a leitora RFID, o circuito e o computador no qual o programa é executado, além de uma pequena porta que servirá para ilustrar a de tamanho normal de uma sala controlada pelo sistema proposto pelo projeto. Na Figura 3.19 pode-se visualizar a porta feita para a maquete.



Figura 3.19 – Porta e fechadura

Além da porta, na maquete usou-se o próprio circuito eletrônico criado para o projeto (Figura 3.20), a leitora RFID (Figura 3.21) e um adaptador TTL RS 232 para o computador visando valer-se dos dados lidos pela leitora (Figura 3.22).

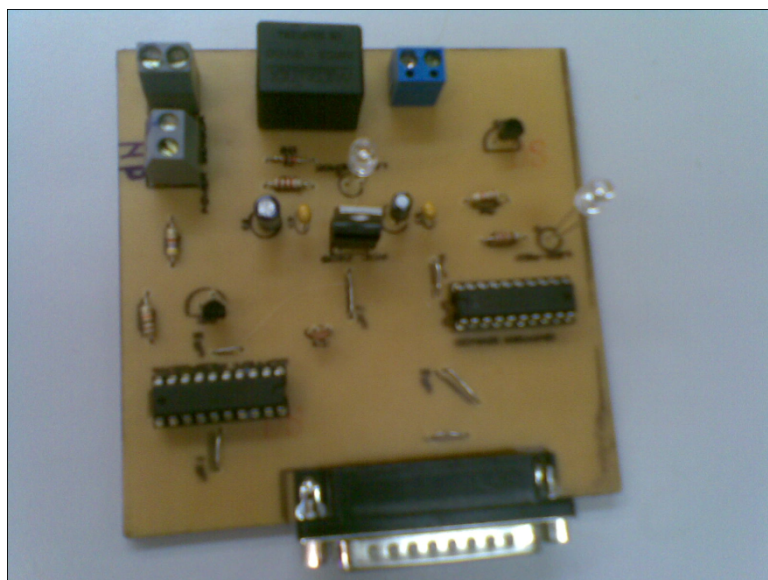


Figura 3.20 – Circuito do projeto



Figura 3.21 – Leitora RFID *Acura Technologies*

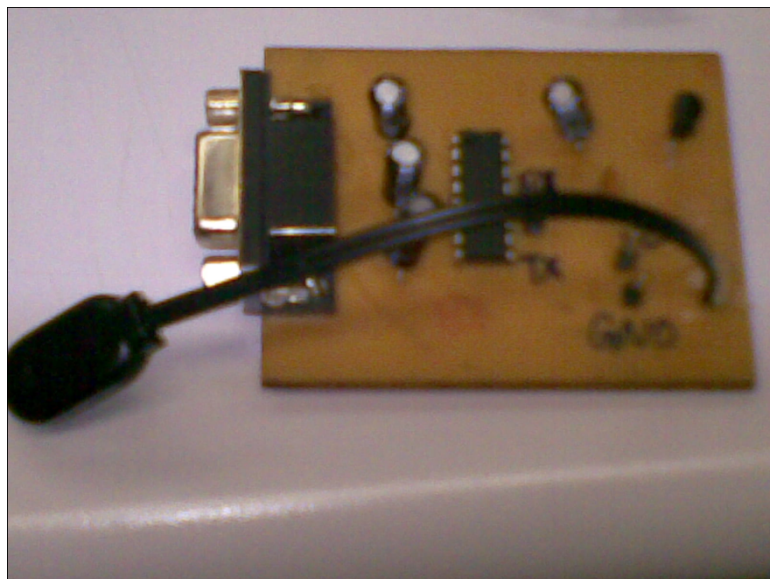


Figura 3.22 – Adaptador TTL RS 232

Após a montagem de toda a estrutura com os dispositivos e a porta, o projeto está pronto para ser demonstrado e executado.

4 CONCLUSÃO

Este trabalho apresenta uma proposta de automação de acesso a empresas e a instituições de ensino, utilizando a tecnologia RFID. Para demonstrar essa automação foi confeccionada uma maquete em escala reduzida, cuja porta ilustrativa contém a fechadura que deverá ser acionada e aberta. No caso de se querer o funcionamento em escala real será preciso somente adequá-lo à necessidade do projeto.

O objetivo principal deste projeto é aumentar a segurança nas dependências de empresas e de instituições de ensino, bem como automatizar a marcação do ponto de presença de funcionários ou de alunos. Para tanto, foi utilizada a tecnologia RFID, a qual agiliza o acesso às dependências da empresa. Foram realizados testes com cartões RFID permitidos e não permitidos, sendo que em nenhuma tentativa com o cartão RFID não permitido a fechadura abriu ou o ponto foi marcado.

Em razão disto, pode-se concluir que o sistema desenvolvido gera mais segurança e mais economia para as empresas, tendo em vista a impossibilidade de marcação indevida de ponto, o que poderia ocasionar horas extras de trabalho, gerando mais custos para a empresa. Se comparado com um produto disponível no mercado, o custo benefício do projeto fica em torno de 30%, uma vez que para desenvolver o projeto foi gasto o valor de R\$ 500,00, o que demonstra que o custo inicial foi relativamente baixo, tornando assim um sistema ideal para o controle de acesso e ponto de presença de funcionários. Através do controle de acesso por RFID, o tempo gasto por funcionário para ingressar na empresa será reduzido consideravelmente e o ponto marcado de forma segura e rápida, o que possibilitará à empresa controlar o horário de entrada e de saída de cada funcionário, registrando a hora, o dia, o mês e o ano.

Quanto às tarefas propostas pelo projeto, obteve-se sucesso na execução quando testadas com o protótipo criado para essa ilustração. A leitora fez a captação do ID do cartão RFID, tendo sido checado de forma satisfatória no banco de dados. Em caso positivo quanto ao cadastro no banco de dados e o envio do sinal para o acendimento do LED verde e o acionamento da abertura da fechadura a tarefa também foi efetuada com sucesso, que se repetiu quando o teste foi referente à não identificação do ID e o envio do sinal para o acendimento do LED vermelho sinalizando a proibição do acesso à dependência da empresa. A programação foi cumprida utilizando C para a execução do envio do sinal para a porta

paralela e PHP para o controle da marcação do ponto e do cadastro de usuários, bem como a leitura do arquivo para a identificação do cartão. O circuito confeccionado para o recebimento do sinal da porta paralela e o acendimento dos LEDs e do relé funcionou de forma perfeita durante todos os testes realizados.

Os resultados obtidos na realização dos testes durante o desenvolvimento do projeto foram satisfatórios do ponto de vista dos objetivos traçados para o projeto. Uma das principais vantagens propostas é a redução do acesso de pessoas que não estejam cadastradas no banco de dados da empresa e também a eliminação do erro na marcação do ponto de entrada e de saída dos funcionários e até mesmo de visitantes que tenham entrado nas dependências da empresa. Porém, vale ressaltar que este trabalho limita-se a estes aspectos e não considera outras possibilidades quanto à segurança ou qualquer situação diferente da então proposta e que possa vir a ocorrer.

O modelo sugerido apresenta diversas vantagens em sua implementação, mesmo levando em conta o custo inicial, como a aquisição da leitora que é a parte mais onerosa do projeto. O RFID pode ter outras utilidades sem necessidade de mudança de tecnologia e de aparelhos, tendo em vista os confrontos expostos com outras tecnologias, tais como o código de barras.

Sendo assim, e levando-se em conta o custo benefício da implementação, o rendimento é consideravelmente satisfatório.

4.1 Problemas Encontrados

Durante a execução do projeto alguns problemas foram detectados.

- Dificuldades com o acesso à porta paralela;
- Dificuldades com o acesso à porta serial;
- Dificuldades com o desenvolvimento de programação para recebimento de sinal da porta serial;
- Dificuldades quanto à aquisição da leitora RFID, principalmente no Brasil;
- Dificuldades com a identificação do sinal criado pela leitora e a marcação no banco de dados.

4.2 Sugestões de Trabalhos Futuros

Com os estudos efetuados para o desenvolvimento do projeto surgiram ideias referentes à possibilidade de implementação de novos trabalhos utilizando o RFID, uma vez que há inúmeras possibilidades de aplicação deste sistema.

São apresentadas sugestões para implantação de novos trabalhos utilizando o RFID:

- Implantação de uma roleta para o acesso de somente uma pessoa por vez, o que impossibilitará a entrada de quem não estiver com permissão para circular pelas dependências de determinada empresa.
- Usar duas leitoras RFID para colocar uma de cada lado da porta.

REFERÊNCIAS

Congresso Brasileiro de Tecnologia, Sistemas e Serviços com RFID.

Em: www.congressorfid.com.br

Acesso em 11/09/2010

Identificação por Rádio Frequência.

Em: <http://radio-frequency.blogspot.com>

Acesso em 27/10/2010

Leonardo4c's Blog.

Em: <http://leonardo4c.files.wordpress.com/>

Acesso em 21/10/2010

MANZANO, José Augusto. MySQL 5 Interativo.

Editora Érica. São Paulo 2007. Página 332.

MECENAS, Ivan; DE OLIVEIRA, Vivianne. Banco de Dados.

Editora Alta Books. Rio de Janeiro 2005. Página 180.

RFID Journal.

Em: <http://www.rfidjournal.com>

Acesso em 10/09/2010

RFID Systems.

Em: <http://www.rfidsystems.com.br/>

Acesso em 26/10/2010

Saber Eletrônica.

Em: <http://www.sabereletronica.com.br/>

Acesso em 26/10/2010

SANGHERA, Paul. RFID+ Study Guide and Practice Exam.

Syngress Publishing. Massachusetts 2007. Página 352.

Sinal Amplificado – Telecomunicações, Redes, Móviles, TI.

Em: <http://www.sinalamplificado.com.br/>

Acesso em 24/10/2010

TEOREY, Toby; LIGHTSTONE, Sam; NADEAU, Tom. Projeto e Modelagem de Bancos de Dados.

Editora Campus/Elsevier. Rio de Janeiro 2007. Página 276.

THORNTON, Frank. RFID Security.
Syngress Publishing. Massachusetts 2006. Página 264.

Univesidade do Porto – Faculdade de Engenharia.
Em: <http://www.fe.up.pt>
Acesso em 23/10/2010

WELLING, Luke; THOMSON, Laura. Tutorial MySQL.
Editora Ciência Moderna. Rio de Janeiro 2004. Página 277.

WIKILIVROS – Livros abertos por um mundo aberto.
Em: <http://pt.wikibooks.org/>
Acesso em 26/10/2010

WIKIPEDIA – A enciclopédia livre
Em: <http://pt.wikipedia.org/>
Acesso em 26/10/2010

ANEXO A: CÓDIGO FONTE COMUNICAÇÃO COM A PORTA PARALELA.

Fonte: ROGERCOM

Função envia dados para pino 8 da porta paralela.

```
#include <stdio.h> // Biblioteca Standard Input/Output
#include <conio.h> // Biblioteca necessária para o uso da função getch();
#include <windows.h> // Biblioteca necessária para o carregamento da inpout32.dll
#include <stdlib.h>
/*Inpout32*/
//Declaração dos ponteiros para função.
typedef short _stdcall (*PtrInp)(short EndPorta);
typedef void _stdcall (*PtrOut)(short EndPorta, short datum);
HINSTANCE hLib; //Instância para a DLL inpout32.dll.
PtrInp inportB; //Instância para a função Imp32(). Para entrada de dados.
PtrOut outportB; //Instância para a função Out32(). Para envio de dados.

/*Inpout32*/
int main()
{
/*Inpout32*/
//Carrega a DLL na memória.
hLib = LoadLibrary("inpout32.dll");
if(hLib == NULL)
{
printf("\n\taErro ao carregar DLL. O arquivo inpout32.DLL nao foi
encontrado.\n");
getch();
}
else { //Todo o programa só será executado apenas se a dll for carregada.
//Obtém o endereço da função Inp32 contida na DLL.
inportB = (PtrInp) GetProcAddress(hLib, "Inp32");
if(inportB == NULL)
```

```

{
    printf("\n\taErro ao tentar obter o endereço da função Inp32.");
}
//Obtém o endereço da função Out32 contida na DLL.
outportB = (PtrOut) GetProcAddress(hLib, "Out32");
if(outportB == NULL)
{
    printf("\n\taErro ao tentar obter o endereço da função Out32.");
}
{
    outportB(0x378,64);//Manda um bit para o pino 8 para acender LED verde e
    relé e abrir porta.
    Sleep(1000);//função para criar um delay para parar de enviar o sinal para o
    pino
    outportB(0x378,0);//Zera a porta paralela para não enviar mais sinal.
}
} //fim do else pertencente ao if que testa se carregou a dll
} //fim da função main e do programa

```

Função envia dados para pino 9 da porta paralela.

Fonte: ROGERCOM

```
#include <stdio.h> // Biblioteca Standard Input/Output
#include <conio.h> // Biblioteca necessária para o uso da função getch();
#include <windows.h> // Biblioteca necessária para o carregamento da inpout32.dll
/*Inpout32*/
//Declaração dos ponteiros para função.
typedef short _stdcall (*PtrInp)(short EndPorta);
typedef void _stdcall (*PtrOut)(short EndPorta, short datum);
HINSTANCE hLib; //Instância para a DLL inpout32.dll.
PtrInp inportB; //Instância para a função Imp32().
PtrOut outportB; //Instância para a função Out32().
/*Inpout32*/
int main()
{
/*Inpout32*/
//Carrega a DLL na memória.
hLib = LoadLibrary("inpout32.dll");
if(hLib == NULL)
{
printf("\n\taErro ao carregar DLL. O arquivo inpout32.DLL nao foi
encontrado.\n");
getch();
}
else {//Todo o programa só será executado apenas se a dll for carregada.
//Obtém o endereço da função Inp32 contida na DLL.
inportB = (PtrInp) GetProcAddress(hLib, "Inp32");
if(inportB == NULL)
{
printf("\n\taErro ao tentar obter o endereço da função Inp32.");
}
//Obtém o endereço da função Out32 contida na DLL.
outportB = (PtrOut) GetProcAddress(hLib, "Out32");
```

```
if(outportB == NULL)
{
    printf("\n\aErro ao tentar obter o endereço da função Out32.");
}

    outportB(0x378,128);//Manda um bit para o pino 9 para negar a entrada.
    Sleep(1000);//função para criar um delay para parar de enviar o sinal para o
pino

    outportB(0x378,0);//Zera a porta paralela para não enviar mais sinal.
} //fim do else pertencente ao if que testa se carregou a dl
} //fim da função main e do programa
```

ANEXO B: PROCESSA PONTO

Processamento do ponto de entrada dos funcionários

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="refresh" content="2;url=processa_ponto.php" />
<title>Processamento do Ponto</title>
</head>
<body>
<body background="images/rfid-tag1.jpg">
<table>
<tr>
<table id="minhaTabela" cellpadding="0" cellspacing="0">
<img src='images/controle2.jpg'>
</tr>
</table>
</table>
<br>
<h1>Processamento do Ponto dos Funcionários</h1>
<h1>Status:</h1>
<br />
<?php
include("conectar.php");
date_default_timezone_set('America/Sao_Paulo');
// Verificar se existe o arquivo
$filename = 'dados/leitorea.txt';
$arquivo = filesize($filename);
if ($arquivo > 1)
{
//abrir o arquivo
$dados = fopen("dados/Leitora.txt", "r");

```

```

$func_tag_bruto = fread($dados, 100);
fclose($dados);

$func_tag = bin2hex($func_tag_bruto);
//verificar se o crachá está cadastrado

$sql = "SELECT func_nome, func_matricula, func_id from tb_funcionarios where
func_tag = " . $func_tag . " ";

$result = mysql_query($sql) or die (mysql_error());
$total = mysql_num_rows($result);
for($i=0; $i<$total; $i++){
    $linha = mysql_fetch_array($result);
    $func_id = $linha["func_id"];
    $func_nome = $linha["func_nome"];
    $func_matricula = $linha["func_matricula"];
    $hora = date(DATE_RFC822);
    echo "</br>$hora</br>";
    echo "</br>Matricula - $func_matricula ";
    echo "</br>Nome - $func_nome </br>";
}

// Se o crachá não for cadastrado mostrar mensagem e executar trava.
if($func_id <1){
    echo"</br>Crachá não cadastrado - Acesso Negado";
    echo"</br>Favor procurar recepção!";
    exec("C:\Documents and Settings\Suporte\Meus documentos\Nega
entrada.exe");
}

// Se o crachá for cadastrado inserir marcação no banco e liberar acesso
else{
    // verificar ultima marcação efetuada
    $sql = "SELECT ponto_id, ponto_tipo from tb_ponto where func_tag
= " . $func_tag . " ";

    $result = mysql_query($sql) or die (mysql_error());
    $total = mysql_num_rows($result);
    for($i=0; $i<$total; $i++){
        $ultimo_id = 1;

```

```

$linha = mysql_fetch_array($result);
$ponto_id = $linha["ponto_id"];
$ponto_tipo = $linha["ponto_tipo"];
    if ($ponto_tipo >= $ultimo_id) {
        $ultimo_id = $ponto_id;
    }
}
//alternar variável entre entrada e saída
switch ($ponto_tipo)
{
    case "":
        $tipo_atual = "Entrada";
        break;
    case "Entrada":
        $tipo_atual = "Saída";
        break;
    case "Saída":
        $tipo_atual = "Entrada";
        break;
}
// inserir marcação no banco
$sql = "INSERT into tb_ponto set ponto_hora = now(), func_tag = "
. $func_tag . ", ponto_tipo = " . $tipo_atual . " ";
$result = mysql_query($sql) or die (mysql_error());
echo "<br>Ponto Processado";
exec("C:\Documents and Settings\Suporte\Meus documentos\Abre
porta.exe");
}
fopen("dados/Leitora.txt","w+");
}

// Caso o arquivo não exista, esperar
else {
    echo "Favor Passar Crachá de Identificação";
}

```

```
include("desconectar.php");
```

```
?>
```

```
</body>
```

```
</html>
```


ANEXO C: CADASTRA, ALTERA E EXCLUI DADOS DE FUNCIONÁRIOS

Programação para cadastrar, alterar e excluir dados de funcionários no banco de dados.

```

<title>Cadastro, Alteração e Exclusão de Funcionários</title>
<?php
include("conectar.php");
//declaração de variáveis.
$operacao = $_POST["operacao"];
$func_id = $_POST["func_id"];
switch ($operacao)
{
//alteração de cadastro de funcionários.
case "Alterar":
    if ($func_id != "")
    {
        $sql = "SELECT func_nome, func_matricula, func_tag, func_cargo from
tb_funcionarios where func_id = " . $func_id . " ";
        $result = mysql_query($sql)
            or die (mysql_error());
        //para recebimento de conteúdo do campos.
        $linha = mysql_fetch_array($result);
        $func_nome = $linha["func_nome"];
        $func_matricula = $linha["func_matricula"];
        $func_tag = $linha["func_tag"];
        $func_cargo = $linha["func_cargo"];
    }
    break;
default:
    $operacao = "Incluir";
    $func_id = "";
    $func_nome = "";
    $func_matricula = "";
    $func_tag = "";

```

```

        $func_cargo = "";
    }
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<body>
<body background="images/rfid-tag1.jpg">
<script language="javascript">
function EfetuaOperacao(id,operacao)
//para confirmação de exclusão de dados.
    if (operacao == 'Excluir')
    {
        if (!confirm('Confirma exclusão do registro?'))
        {
            return
        }
    }
    if (operacao == 'Alterar')//alteração de dados dos funcionários.
    {
        document.cadastro.action="func_form.php";
    }
    else
    {
        document.cadastro.action="func_commit.php";
    }
    document.cadastro.func_id.value=id;
    document.cadastro.operacao.value=operacao;
    document.cadastro.submit();
}
</script>

```

```

</head>
<body>
<?php include("menu.php"); ?>
<table id="minhaTabela" cellpadding="1" cellspacing="1">
    <tr>
<td colspan="5" id="titulo"><strong>Cadastro de Funcionários</strong></td>
    </tr>
    <tr id="cabecalho">
        <td id="func_nome"><strong>Nome</strong></td>
        <td id="func_matricula"><strong>Matrícula</strong></td>
        <td id="func_tag"><strong>Crachá</strong></td>
        <td id="func_cargo"><strong>Cargo</strong></td>
        <td id="Operação"><strong>Operação</strong></td>
    </tr>

<?php
echo "<h1>" . $operacao . " Funcionário</h1>"; //exibe a caixa com os campos para
preenchimento do cadastro.
echo "<hr><br>";
echo "    <form    name='cadastro'    action='func_commit.php'    method='post'
enctype='multipart/form-data'>";
echo "<input name='operacao' type='hidden' value='" . $operacao . "'>";
echo "<input name='func_id' type='hidden' value='" . $func_id . "'>";

echo "<td class=\"linhas\"><input name='func_nome' type='text' size='55' value='" .
$func_nome . "'></td>";
echo "    <td    class=\"linhas\"><input    name='func_matricula'    type='text'    size=10
value='" . $func_matricula . "'></td>";
echo "    <td    class=\"linhas\"><input    name='func_tag'    type='text'    size=40    value='" .
$func_tag . "'></td>";
echo "    <td    class=\"linhas\"><input    name='func_cargo'    type='text'    size=15    value='" .
$func_cargo . "'></td>";
echo "    <td    class=\"linhas\"><input type='submit' value='" . $operacao . "'></td>";
echo "</form>";
?>

```

```

</tr>
</table>
<div>
<br>
<br />
<table id="minhaTabela" cellpadding="1" cellspacing="1">
    <tr>
        <td colspan="7" id="titulo"><strong>Funcionários</strong></td>
    </tr>
    <tr id="cabecalho">
        <td id="nome"><strong>Nome</strong></td>
        <td id="matricula"><strong>Matrícula</strong></td>
        <td id="cracha"><strong>Crachá</strong></td>
        <td id="cargo"><strong>Cargo</strong></td>
        <td colspan="2" id="Operação"><strong>Operação</strong></td>
    </tr>

<?php
//exibe os funcionários cadastrados, bem como as opções de excluir e alterar.
$sql = "SELECT func_id, func_nome, func_matricula, func_tag, func_cargo from
tb_funcionarios";

$result = mysql_query($sql) or die (mysql_error());
$total = mysql_num_rows($result);
for($i=0; $i<$total; $i++){
    $linha = mysql_fetch_array($result);
    $func_id = $linha["func_id"];
    $func_nome = $linha["func_nome"];
    $func_matricula = $linha["func_matricula"];
    $func_tag = $linha["func_tag"];
    $func_cargo = $linha["func_cargo"];
    echo "<tr id='$func_id'>";
    echo "<td class='linhas'>$func_nome</td>";
    echo "<td class='linhas'>$func_matricula</td>";
    echo "<td class='linhas'>$func_tag</td>";
    echo "<td class='linhas'>$func_cargo</td>";

```

```
echo "<td class='linhas'><a href='#' onclick=\"EfetuaOperacao('$func_id','Alterar');\"><img  
src='images/editar.gif' alt='Editar' title='Editar'></a></td>";  
echo "<td class='linhas'><a href='#' onclick=\"EfetuaOperacao('$func_id','Excluir');\"><img  
src='images/excluir.gif' alt='Excluir' title='Excluir'></a></td>";  
}  
?>  
<tr>  
</tr>  
</table>  
</div>  
</br>  
</body>  
</html>
```

ANEXO D: RELATÓRIO DE PONTOS

Programação do menu para impressão de relatório de ponto dos funcionários

```

<title>Cadastro, Alteração e Exclusão de Funcionários</title>
<?php
include("conectar.php");
//declaração de variáveis.
$operacao = $_POST["operacao"];
$func_id = $_POST["func_id"];
switch ($operacao)
{
//alteração de cadastro de funcionários.
    case "Alterar":
        if ($func_id != "")
        {
            $sql = "SELECT func_nome, func_matricula, func_tag, func_cargo from
tb_funcionarios where func_id = " . $func_id . " ";
            $result = mysql_query($sql)
                or die (mysql_error());
            //para recebimento de conteúdo do campos.
            $linha = mysql_fetch_array($result);
            $func_nome = $linha["func_nome"];
            $func_matricula = $linha["func_matricula"];
            $func_tag = $linha["func_tag"];
            $func_cargo = $linha["func_cargo"];
        }
        break;
    default:
        $operacao = "Incluir";
        $func_id = "";
        $func_nome = "";
        $func_matricula = "";
        $func_tag = "";

```

```

        $func_cargo = "";
    }
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<body>
<body background="images/rfid-tag1.jpg">
<script language="javascript">
function EfetuaOperacao(id,operacao)
//para confirmação de exclusão de dados.
    if (operacao == 'Excluir')
    {
        if (!confirm('Confirma exclusão do registro?'))
        {
            return
        }
    }
    if (operacao == 'Alterar')//alteração de dados dos funcionários.
    {
        document.cadastro.action="func_form.php";
    }
    else
    {
        document.cadastro.action="func_commit.php";
    }
    document.cadastro.func_id.value=id;
    document.cadastro.operacao.value=operacao;
    document.cadastro.submit();
}
</script>

```

```

</head>
<body>
<?php include("menu.php"); ?>
<table id="minhaTabela" cellpadding="1" cellspacing="1">
    <tr>
        <td colspan="5" id="titulo"><strong>Cadastro de
Funcionários</strong></td>
    </tr>
    <tr id="cabecalho">
        <td id="func_nome"><strong>Nome</strong></td>
        <td id="func_matricula"><strong>Matrícula</strong></td>
        <td id="func_tag"><strong>Crachá</strong></td>
        <td id="func_cargo"><strong>Cargo</strong></td>
        <td id="Operação"><strong>Operação</strong></td>
    </tr>

<?php
echo "<h1> . $operacao . " Funcionário</h1>"; //exibe a caixa com os campos para
preenchimento do cadastro.
echo "<hr><br>";
echo "<form name='cadastro' action='func_commit.php' method='post'
enctype='multipart/form-data'>";
echo "<input name='operacao' type='hidden' value='\" . $operacao . \"'>";
echo "<input name='func_id' type='hidden' value='\" . $func_id . \"'>";
echo "<td class='\"linhas\"><input name='func_nome' type='text' size='55' value='\" .
$func_nome . \"'></td>";
echo "<td class='\"linhas\"><input name='func_matricula' type='text' size=10
value='\" . $func_matricula . \"'></td>";
echo "<td class='\"linhas\"><input name='func_tag' type='text' size=40 value='\" .
$func_tag . \"'></td>";
echo "<td class='\"linhas\"><input name='func_cargo' type='text' size=15 value='\" .
$func_cargo . \"'></td>";
echo "<td class='\"linhas\"><input type='submit' value='\" . $operacao . \"'></td>";
echo "</form>";
?>

```



```

</tr>
</table>
<div>
<br>
<br />
<table id="minhaTabela" cellpadding="1" cellspacing="1">
    <tr>
        <td colspan="7" id="titulo"><strong>Funcionários</strong></td>
    </tr>
    <tr id="cabecalho">
        <td id="nome"><strong>Nome</strong></td>
        <td id="matricula"><strong>Matrícula</strong></td>
        <td id="cracha"><strong>Crachá</strong></td>
        <td id="cargo"><strong>Cargo</strong></td>
        <td colspan="2" id="Operação"><strong>Operação</strong></td>
    </tr>

<?php
//exibe os funcionários cadastrados, bem como as opções de excluir e alterar.
$sql = "SELECT func_id, func_nome, func_matricula, func_tag, func_cargo from
tb_funcionarios";

$result = mysql_query($sql) or die (mysql_error());
$total = mysql_num_rows($result);
for($i=0; $i<$total; $i++){
    $linha = mysql_fetch_array($result);
    $func_id = $linha["func_id"];
    $func_nome = $linha["func_nome"];
    $func_matricula = $linha["func_matricula"];
    $func_tag = $linha["func_tag"];
    $func_cargo = $linha["func_cargo"];
    echo "<tr id='$func_id'>";
    echo "<td class='linhas'>$func_nome</td>";
    echo "<td class='linhas'>$func_matricula</td>";
    echo "<td class='linhas'>$func_tag</td>";

```

```

echo "<td class='linhas'>$func_cargo</td>"; echo "<td class='linhas'><a href='#'
onclick=\"EfetuaOperacao('$func_id','Alterar');\"><img src='images/editar.gif' alt='Editar'
title='Editar'></a></td>";
echo "<td class='linhas'><a href='#' onclick=\"EfetuaOperacao('$func_id','Excluir');\"><img
src='images/excluir.gif' alt='Excluir' title='Excluir'></a></td>";
    }
?>
<tr>
</tr>
</table>
</div>
</br>
</body>
</html>

```

ANEXO E: RELATÓRIO POR FUNCIONÁRIO

Programação de impressão de relatório por nome de funcionário

```

<?php
include("menu.php"); //inclusão do menu e conexão com o banco de dados.
include("conectar.php");
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Relatório de Ponto por Funcionário</title>
</head>
<body>
<body background="images/rfid-tag1.jpg">
<div>
<br>
<h1>Relatório de Ponto por Funcionários</h1>
<br />
<?php
echo "$func_id";

//select para a coleta de dados dos funcionarios no banco e seus pontos..
$sql="SELECT FC.func_nome as nome, FC.func_cargo as cargo, FC.func_matricula
as matricula, FC.func_id as func_id, PT.ponto_hora as hora, PT.ponto_tipo as tipo
from tb_funcionarios as FC
LEFT JOIN tb_ponto as PT
ON FC.func_tag = PT.func_tag
";

$result = mysql_query($sql) or die (mysql_error());
$total = mysql_num_rows($result);

```

```

$func_atual= 0;
for ($i=0;$i<$total;$i++)
{
    //declaração das variáveis.
    $linha = mysql_fetch_array($result);
    $func_id = $linha["func_id"];
    $func_matricula = $linha["matricula"];
    $func_cargo = $linha["cargo"];
    $func_nome = $linha["nome"];
    $func_tag = $linha["tag"];
    $ponto_hora = $linha["hora"];
    $ponto_tipo = $linha["tipo"];
    if ($i==0)//exibição do relatório dos pontos marcados.
    {
        echo"<table id='minhaTabela' cellpadding='1' cellspacing='1'> ";
        echo"<tr>
        <td colspan='3' id='titulo'><strong>$func_nome - $func_cargo</strong></td>
        </tr>";
        echo"<tr id='cabecalho'>
            <td id='pc_id'><strong>Dia</strong></td>
            <td id='pc_trecho'><strong>Tipo</strong></td>
        </tr>";
        $func_atual=$func_id;
    }
    //exibição do próximo funcionário cadastrado.
    elseif ($func_id!= $func_atual)
    {
        echo "<tr>";
        echo"<table id='minhaTabela' cellpadding='1' cellspacing='1'> ";
        echo"<tr>
        <td colspan='3' id='titulo'><strong>$func_nome - $func_cargo</strong></td>
        </tr>";
        echo"<tr id='cabecalho'>
            <td id='pc_id'><strong>Dia</strong></td>
            <td id='pc_trecho'><strong>Tipo</strong></td>
        </tr>";
    }
}

```

```
$func_atual=$func_id;
}
echo "<tr id='$func_id'>";
echo "<td class='linhas'>$ponto_hora</td>";
echo "<td class='linhas'>$ponto_tipo</td>";
}
?>
<tr>
</tr>
</table>
</div>
</br>
</body>
</html>
```

ANEXO F: RELATÓRIO POR DATA

Programação de impressão de relatório por data.

```

<?php
include("menu.php"); //inclusão de menu na página.
include("conectar.php"); // conexão com o banco.
?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<title>Relatório de Ponto de Funcionário por Data</title>
</head>
<body>
<body background="images/rfid-tag.jpg">
<div>
<br>
<h1>Relatório de Ponto por Data</h1>
<br />
<?php
echo "$func_id";

//select para recolher os dados do banco para a exibição do relatório dos pontos.
$sql="SELECT FC.func_nome as nome, FC.func_cargo as cargo, FC.func_matricula
as matricula, FC.func_id as func_id, PT.ponto_hora as marcacao, PT.ponto_tipo as tipo
from tb_funcionarios as FC
LEFT JOIN tb_ponto as PT
ON FC.func_tag = PT.func_tag
Order by PT.ponto_hora, FC.func_id
";

$result = mysql_query($sql) or die (mysql_error());

```

```

$total = mysql_num_rows($result);
$dia_atual= 0;
for ($i=0;$i<$total;$i++)
{
    //declaração das variáveis com os respectivos conteúdos.
    $linha = mysql_fetch_array($result);
    $func_id = $linha["func_id"];
    $func_matricula = $linha["matricula"];
    $func_cargo = $linha["cargo"];
    $func_nome = $linha["nome"];
    $func_tag = $linha["tag"];
    $marcacao = $linha["marcacao"];
    $ponto_tipo = $linha["tipo"];
    $data = split(" ", $marcacao);
    $dia = $data[0];
    $hora = $data[1];
    if ($i==0)
    {
        echo "<table id='minhaTabela' cellpadding='1' cellspacing='1'> ";
        echo "<tr>
        <td colspan='4' id='titulo'><strong>$dia</strong></td>
        </tr>";
        echo "<tr id='cabecalho'>
        <td id='hora'><strong>Hora</strong></td>
        <td id='funcionario'><strong>Funcionario</strong></td>
        <td id='tipo'><strong>Tipo</strong></td>
        </tr>";
        $dia_atual=$dia;
    }
    elseif ($dia!= $dia_atual)
    {
        echo "<tr>";
        echo "<table id='minhaTabela' cellpadding='1' cellspacing='1'> ";
        echo "<tr>
        <td colspan='4' id='titulo'><strong>$dia</strong></td>

```

```

</tr>";
echo "<tr id='cabecalho'>
    <td id='hora'><strong>Hora</strong></td>
    <td id='funcionario'><strong>Funcionario</strong></td>
    <td id='tipo'><strong>Tipo</strong></td>
</tr>";
$dia_atual=$dia;
}
//Para a exibição do relatório conforme o dia.
echo "<tr id='$dia'>";
echo "<td class='linhas'>$hora</td>";
echo "<td class='linhas'>$func_nome</td>";
echo "<td class='linhas'>$ponto_tipo</td>";
}
?>
<tr>
</tr>
</table>
</div>
</br>
</body>
</html>

```


ANEXO G: MENU

Programação da exibição do menu

```

<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<body>
<body background="images/rfid-tag1.jpg">
</head>
<table>
<tr>
<table id="foto menu" cellpadding="0" cellspacing="0">
<table border=2 background="images/rfid-tag1.jpg">
<table id="minhaTabela" cellpadding="0" cellspacing="0">
<img src='images/controle.jpg'>
<td id="titulo"><a href="func_form.php"><strong>Cadastro de
Funcionários</strong></a></td>
<td id="titulo"><a href="rel_form.php"><strong>Relatório de Ponto</strong></td>
<td id="titulo"><a href="ponto_form.php"><strong>Processamento de
Ponto</strong></td>
</tr>
</table>
</table>
<BR>

```

ANEXO H: MARCAÇÃO DO PONTO DO FUNCIONÁRIO

Programação para a marcação do ponto do funcionário

```

<title>Marcação de Ponto de Funcionário</title>
<?php
include("conectar.php");
//declaração das variáveis.
$operacao = $_POST["operacao"];
$func_id = $_POST["func_id"];
switch ($operacao)
{
//para a marcação do ponto.
case "Alterar":
    if ($func_id != "")
    {
        $sql = "SELECT func_nome, func_matricula, func_tag, func_cargo from
tb_funcionarios where func_id = " . $func_id . " ";
        $result = mysql_query($sql)
        or die (mysql_error());
        $linha = mysql_fetch_array($result);
        $func_nome = $linha["func_nome"];
        $func_matricula = $linha["func_matricula"];
        $func_tag = $linha["func_tag"];
        $func_cargo = $linha["func_cargo"];
    }
    break;
default:
    $operacao = "Incluir";
    $func_id = "";
    $func_nome = "";
    $func_matricula = "";
    $func_tag = "";
    $func_cargo = "";

```

```

    }
    ?>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<link rel="stylesheet" type="text/css" href="estiloDataGrid.css">
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
<script language="javascript">
function EfetuaOperacao(id,operacao)
{
    if (operacao == 'Excluir')
    {
        if (!confirm('Confirma exclusão do registro?'))
        {
            return
        }
    }
    if (operacao == 'Alterar')
    {
        document.cadastro.action="func_form.php";
    }
    else
    {
        document.cadastro.action="func_commit.php";
    }
    document.cadastro.func_id.value=id;
    document.cadastro.operacao.value=operacao;
    document.cadastro.submit();
}
</script>
<script language="JavaScript">
function moveRelogio(){
    momentoAtual = new Date()

```

```

        hora = momentoAtual.getHours()
        minuto = momentoAtual.getMinutes()
        segundo = momentoAtual.getSeconds()
        horaImprimivel = hora + " : " + minuto + " : " + segundo
        document.form_relogio.relogio.value = horaImprimivel
        setTimeout("moveRelogio()",1000)
    }
</script>
</head>
<body onload="moveRelogio()">
<?php include("menu.php"); ?>
<h1>Hora do Dia</h1>
<form name="form_relogio">
<input type="text" name="relogio" size="6">
</form>
<?php
echo "<h1>Processamento de Ponto</h1>";
echo "<hr><br>";
echo      "<form      name='cadastro'      onSubmit='return      EfetuaOperacao()'
action='processa_ponto.php' method='post' enctype='multipart/form-data'>";
echo "<td class=\"linhas\"><input type='submit' value='Processar Ponto'></td>";
echo "</form>";
?>
<table id="minhaTabela" cellpadding="1" cellspacing="1">
    <tr>
        <td colspan="5" id="titulo"><strong>Marcação de
Ponto</strong></td>
    </tr>
    <tr id="cabecalho">
        <td id="func_tag"><strong>Crachá</strong></td>
        <td id="func_nome"><strong>Nome</strong></td>
        <td id="func_matricula"><strong>Matrícula</strong></td>
        <td id="ponto_tipo"><strong>Tipo</strong></td>
        <td id="Operação"><strong>Tipo</strong></td>
    </tr>
</table>

```

```

        </tr>

<?php
echo "<hr><br>";
echo "    <form    name='cadastro'    action='ponto_commit.php'    method='post'
enctype='multipart/form-data'>";
echo "<input name='hora' type='hidden' value='" . $hora . "'>";
echo "<input name='func_id' type='hidden' value='" . $func_id . "'>";
echo "<td class=\"linhas\"><input name='func_tag' type='text' size=40 value='" .
$func_tag . "'></td>";
echo "<td class=\"linhas\"><input name='func_nome' type='text' size='60' value='" .
$func_nome . "'></td>";
echo "    <td class=\"linhas\"><input name='func_matricula' type='text' size=10
value='" . $func_matricula . "'></td>";
echo "<td class=\"linhas\"><input name='ponto_tipo' type='text' size=10 value='" .
$ponto_tipo . "'></td>";
echo "<td class=\"linhas\"><input type='submit' value=Marcar></td>";
echo "</form>";
?>

</tr>

</table>

</body>

</html>

```

ANEXO I: CONEXÃO E DESCONEXÃO COM O BANCO DE DADOS

Programação para conexão e desconexão com o banco de dados para consulta, alteração e exclusão de dados.

Conexão

```
<?php
```

```
//conexão com o banco de dados.
```

```
$conexao = mysql_connect("localhost", "root", "")
```

```
or die ("Configuração de Banco de Dados Errada!");
```

```
$db = mysql_select_db("ponto")
```

```
or die ("Banco de Dados Inexistente!");
```

```
?>
```

Desconexão

```
<?php
```

```
//fim de conexão com o banco de dados.
```

```
$conexao = mysql_close($conexao)
```

```
or die ("Erro ao fechar conexao com banco de dados!");
```

```
?>
```